

## **Bioinformática e Biofísica Molecular Computacional**

*Introdução à Dinâmica Molecular*

**Luís Fernando Saraiva**

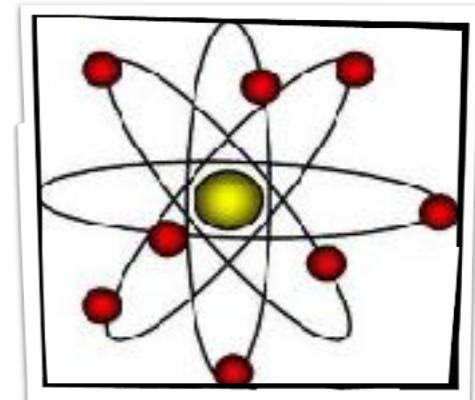
**[luis.timmers@univates.br](mailto:luis.timmers@univates.br)**

# Introdução à Dinâmica Molecular

- ✓ Uma das técnicas computacionais mais versáteis para o estudo de macromoléculas biológicas;
- ✓ Útil em várias etapas do **SBDD**;
- ✓ Metodologia fundamentada nos princípios da **Mecânica Clássica**;
- ✓ Fornece informações sobre o **comportamento dinâmico microscópico, dependente do tempo**, dos átomos individuais que compõem o sistema;
- ✓ Para se obter as **propriedades macroscópicas** de interesse (pressão, energia interna, volume, temperatura, entropia, energia livre...), é necessário utilizar a **Mecânica Estatística**.

# Mecânica Molecular

- ✓ Calcula a ESTRUTURA e ENERGIA das moléculas com base nos movimentos dos **núcleos**;
- ✓ Os **elétrons** não são considerados explicitamente: é assumido que encontrarão uma distribuição ótima, uma vez que as posições dos núcleos são conhecidas.



## Aproximação de Born-Oppenheimer:

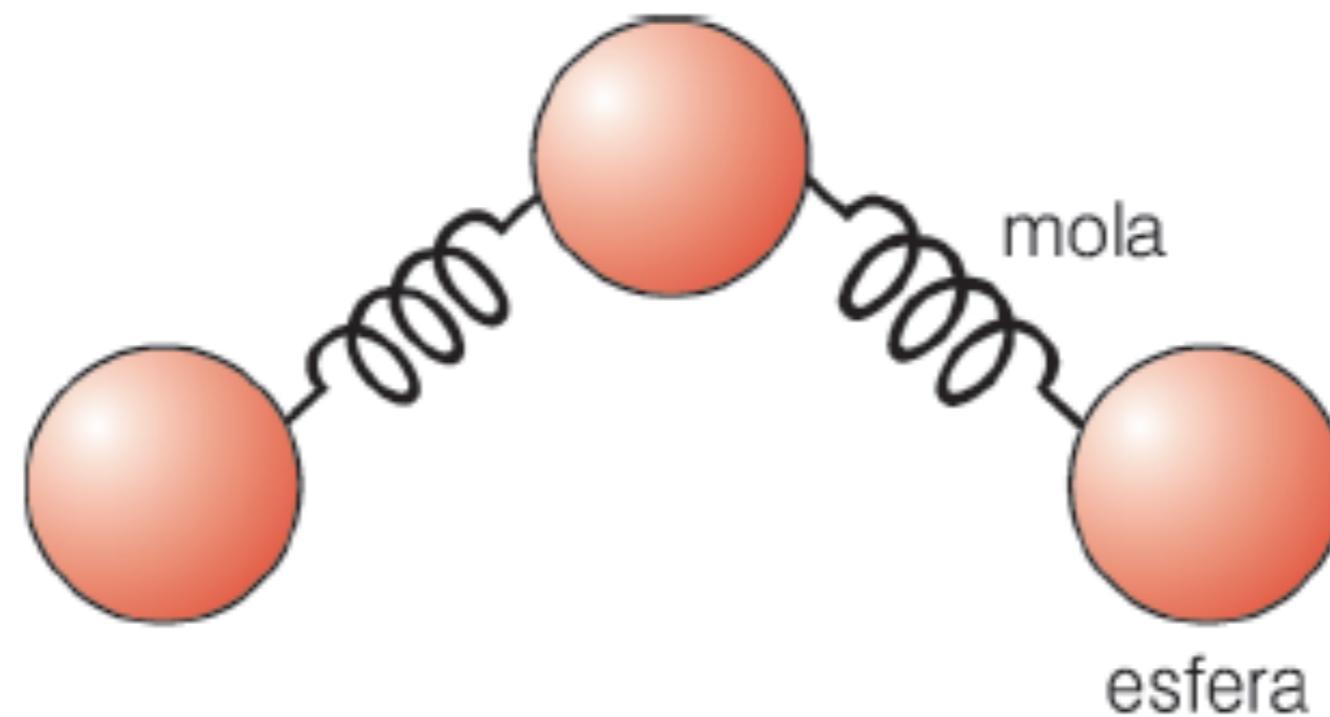
Núcleos + pesados → movem-se mais lentamente → seus movimentos (vibrações e rotações) podem ser estudadas separadamente → elétrons movem-se rapidamente, ajustando-se aos movimentos dos núcleos.

# Mecânica Molecular

A molécula é tratada como uma coleção de esferas conectadas por molas.

Esferas = núcleos (átomos);

Molas = ligações.



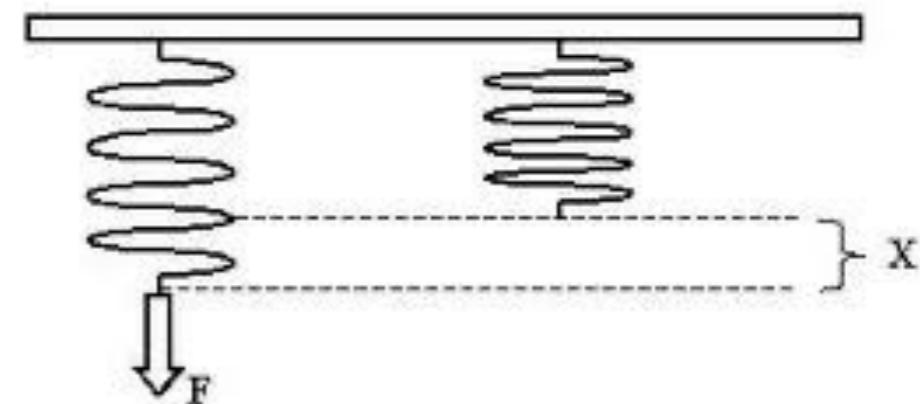
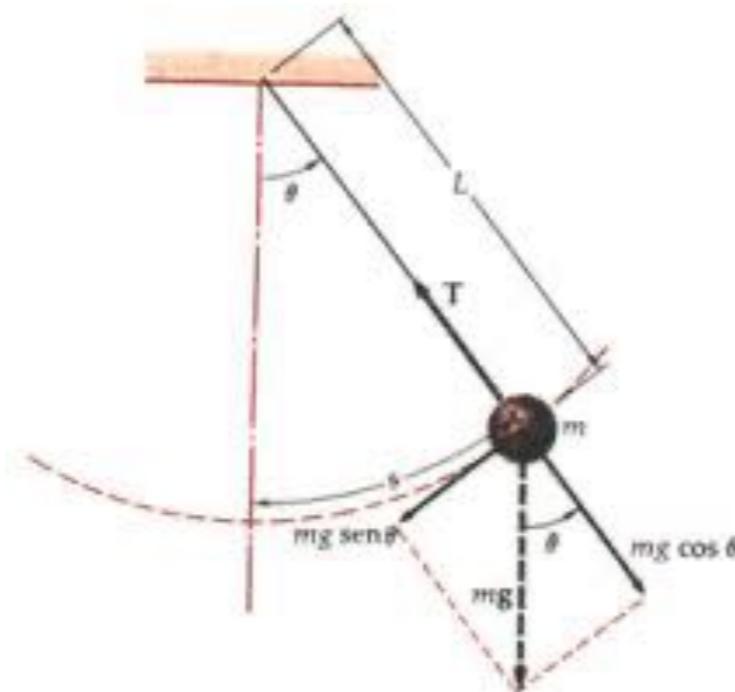
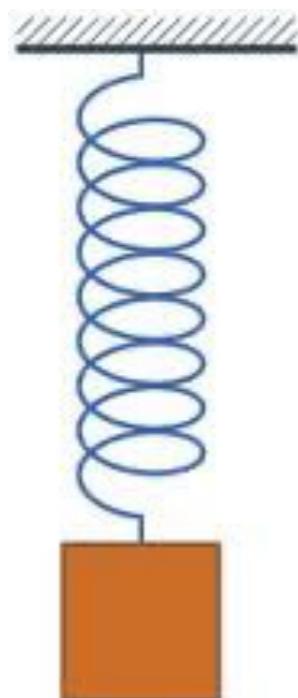
# Mecânica Molecular

- ✓ As moléculas são tratadas como uma coleção de átomos que pode ser descrita por FORÇAS NEWTONIANAS.

Uma coleção de partículas é mantida unida por:

- **Forças harmônicas (oscilador harmônico) e força elástica:**

Comprimento de ligação, ângulo de ligação, ângulo diedro...



$$\text{Lei de Hooke:} \\ F = k X$$

## Campo de Força

- ✓ Conjunto completo dos potenciais de interação entre as partículas, que permite calcular a energia e geometria de uma molécula;
- ✓ É elaborado de forma que contenha:
  - Uma coleção de diferentes tipos de átomos;
  - Parâmetros (para comprimento e ângulos de ligação, etc.);
  - Equações para calcular a energia de uma molécula.
- ✓ Os parâmetros são provenientes de dados experimentais ou cálculos de mecânica quântica.

## Campo de Força

Um típico Campo de Força é representado pela Equação:

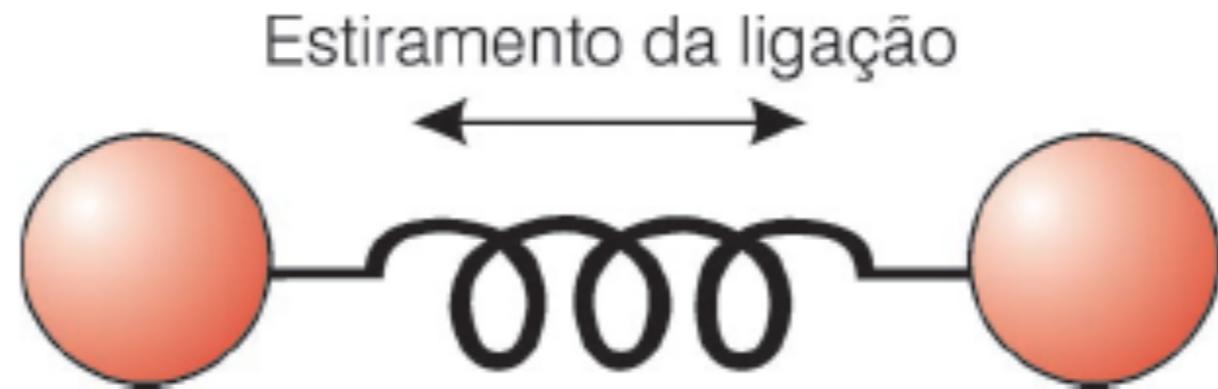
$$V(r) = \sum V_l + \sum V_o + \sum V_{vdW} + \sum V_{elet}$$

Onde:

**$V_l$ :** energia de estiramento da ligação em relação ao seu valor de equilíbrio (ideal);

# Campo de Força

## Deformação no comprimento da ligação



Estiramento ou compressão da ligação = Aumento de energia da molécula

## Campo de Força

Um típico Campo de Força é representado pela Equação:

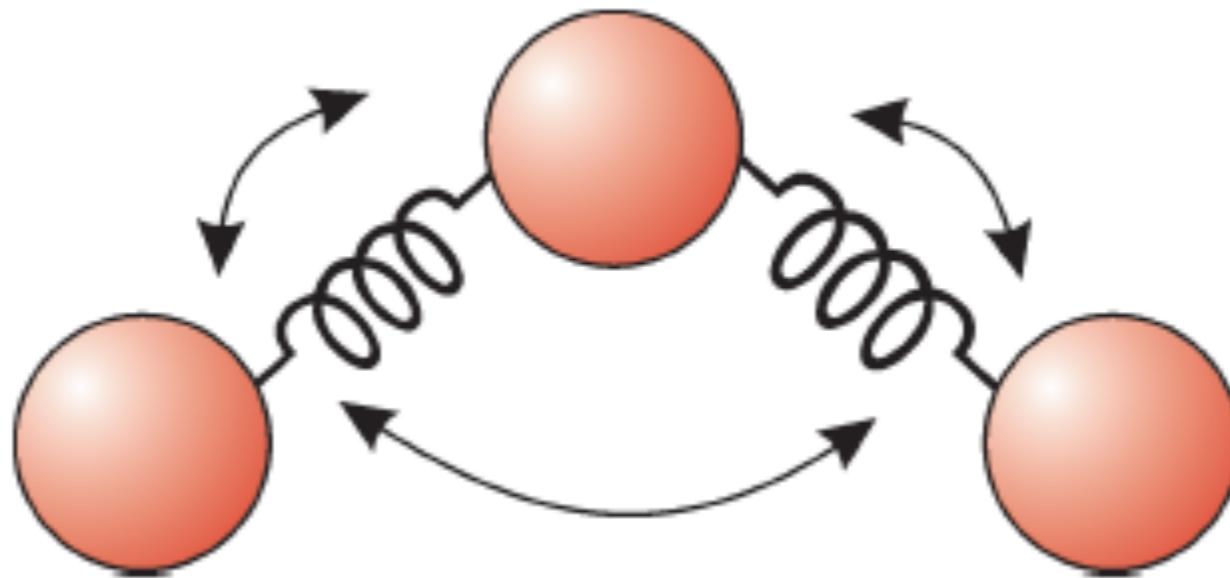
$$V(r) = \sum V_l + \sum V_\theta + \sum V_{vdW} + \sum V_{elet}$$

**onde:**

**$V_\theta$ :** energia de deformação do ângulo de ligação em  
relação ao seu valor de equilíbrio (ideal);

# Campo de Força

## Deformação angular



Em relação a diferentes tipos de átomos e hibridizações

## Campo de Força

Um típico Campo de Força é representado pela Equação:

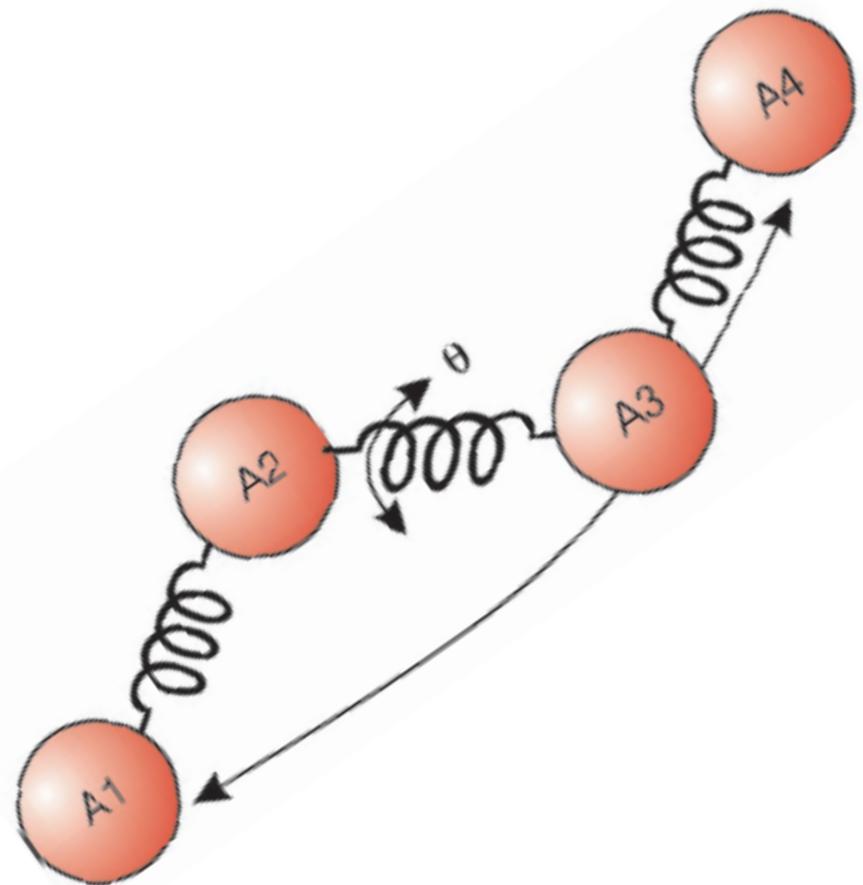
$$V(r) = \sum V_l + \sum V_\phi + \sum V_{vdW} + \sum V_{elet}$$

**onde:**

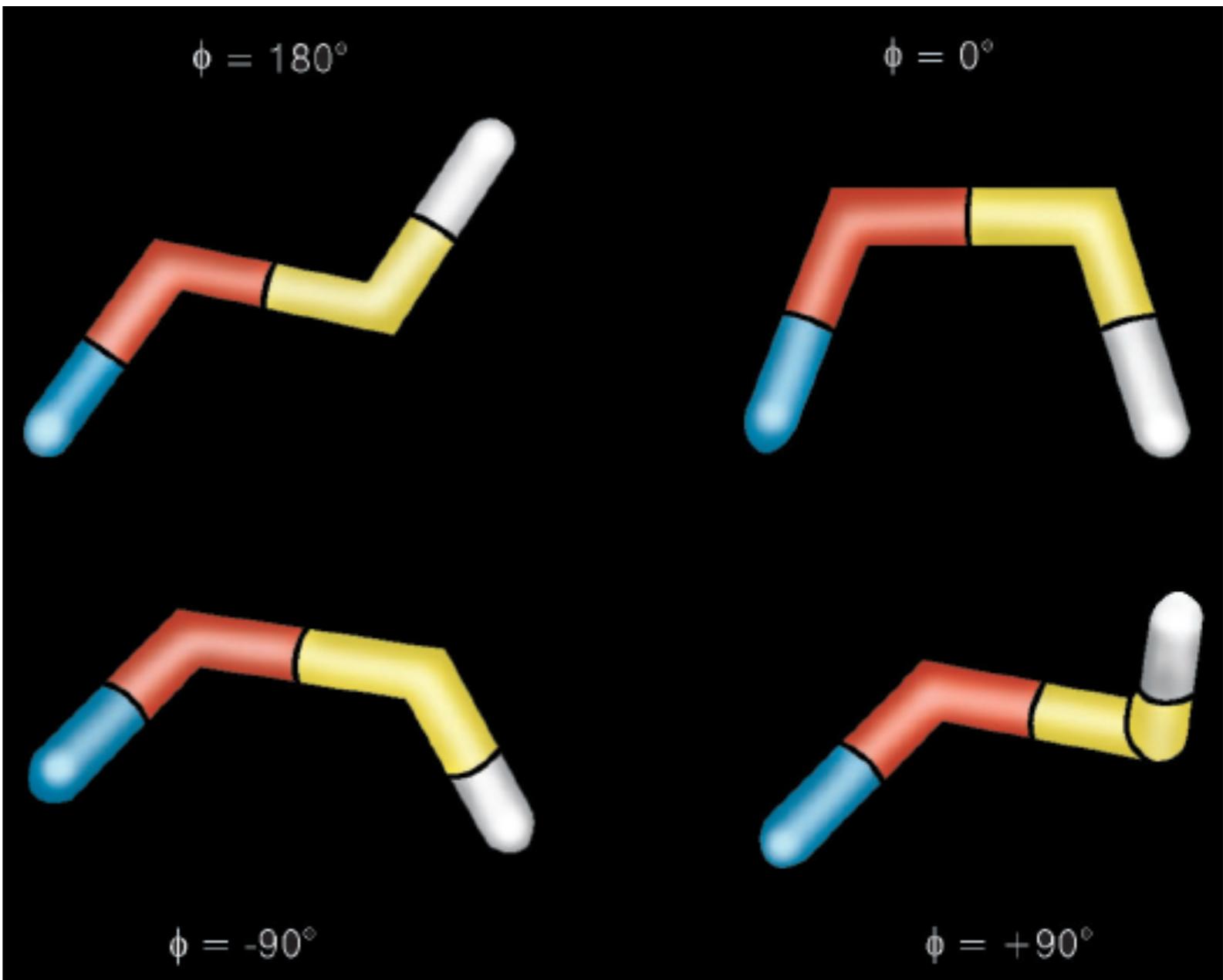
$V_\phi$ : energia devido à torção em torno de uma ligação em relação ao seu valor ideal;

# Campo de Força

Barreira de energia de interações intramoleculares (ângulos de torção)



Representação do ângulo de diedro delimitados por 4 átomos consecutivos, assinalando diferentes valores de ângulo diedro.



## Campo de Força

Um típico Campo de Força é representado pela Equação:

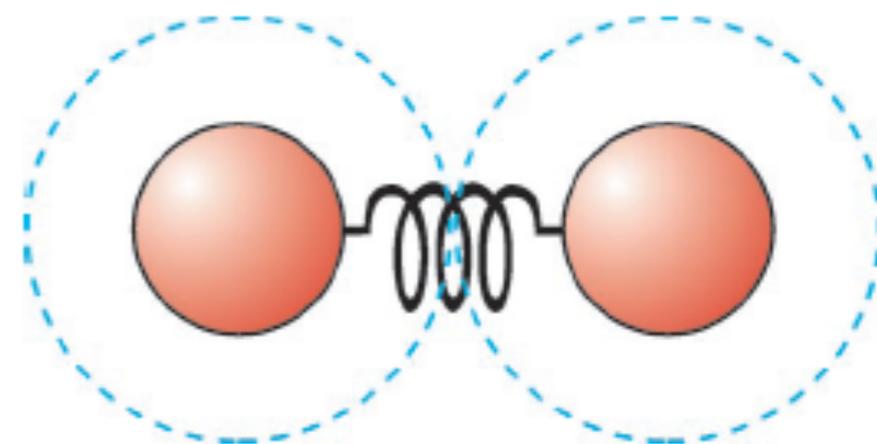
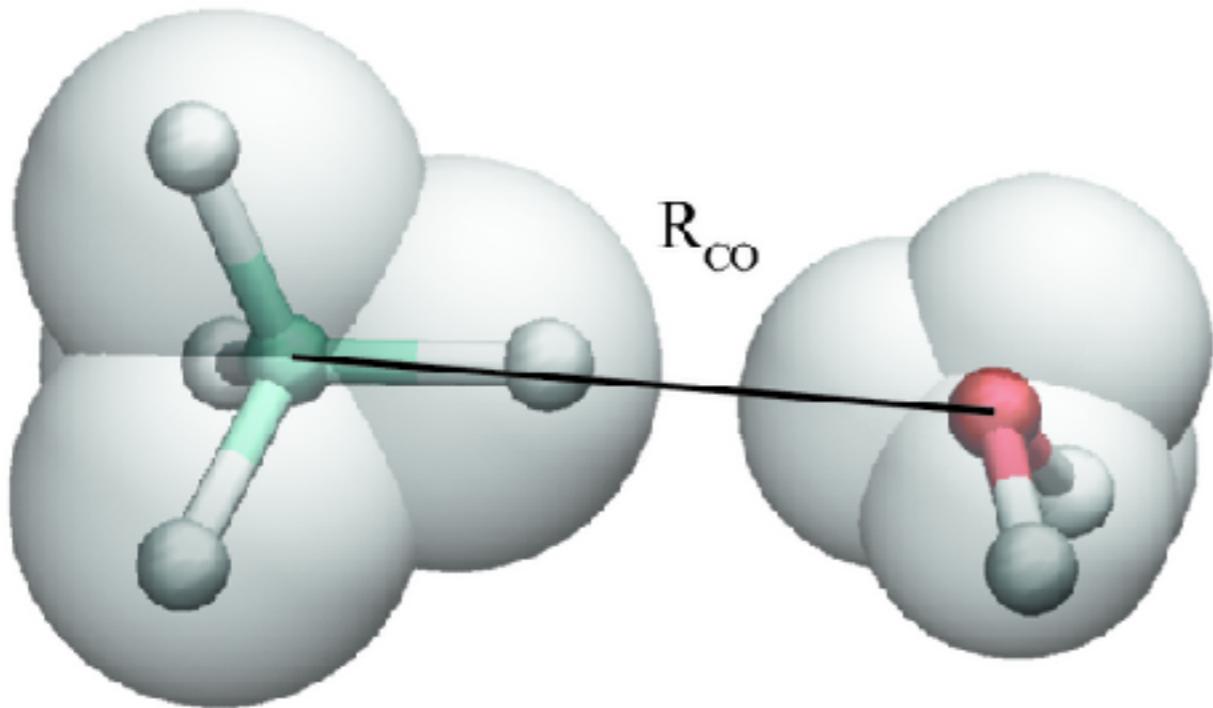
$$V(r) = \sum V_l + \sum V_o + \sum V_{vdW} + \sum V_{elet}$$

**onde:**

**$V_{vdW}$ :** representa a energia das interações de van der Waals;

# Campo de Força

## Interações de van der Waals



*Raio de vdW = tamanho efetivo do átomo;*

*Aproximação entre átomos não ligados =  $\uparrow$  da interação de vdW =  $\downarrow$  energia;*

*Distância entre dois átomos = soma dos seus raios vdW = atração máxima;*

*Maior aproximação = forte repulsão vdW.*

## Campo de Força

Um típico Campo de Força é representado pela Equação:

$$V(r) = \sum V_l + \sum V_o + \sum V_{vdW} + \sum V_{elet}$$

**onde:**

**$V_{elet}$ :** representa a energia de atração ou repulsão eletrostática entre duas cargas.

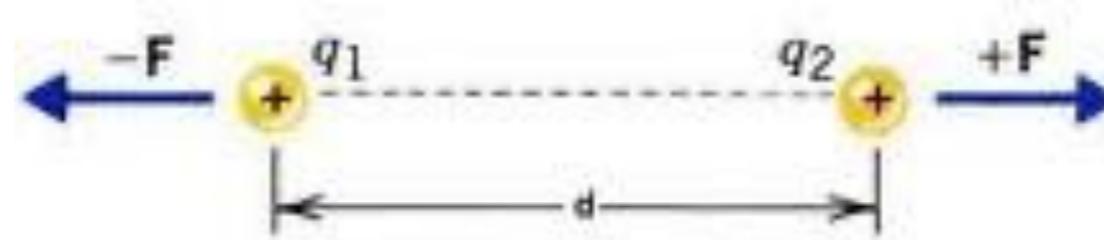
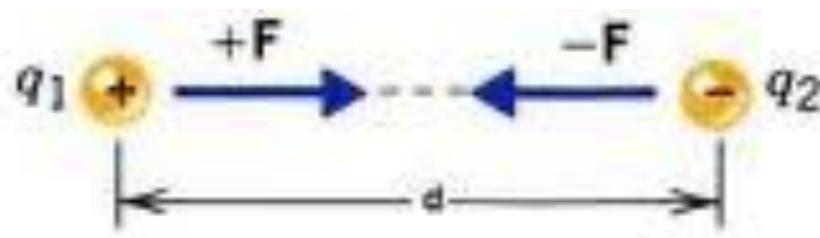
## Campo de Força

Força de interação entre duas partículas – **Lei de Coulomb**

$$F = k \frac{q_1 q_2}{d^2}$$

K = 9 . 10<sup>9</sup> unidades do SI

A **interação eletrostática** entre partículas eletrizadas manifesta-se por meio de forças de atração e repulsão, dependendo dos sinais das cargas:



## Campo de Força

Além destes existem muitos outros parâmetros...

- ✓ Os campos de forças existentes foram desenvolvidos de maneira independente e com todos os conjuntos de parâmetros específicos;
- ✓ Alguns incluem parâmetros específicos para determinado tipo de interação;
- ✓ A confiabilidade dos resultados da DM é baseada na elaboração de um campo de força com parâmetros bem definidos;
- ✓ A escolha do campo de força depende, em grande parte, do sistema a ser estudado e das propriedades que serão investigadas;
- ✓ No caso de sistemas biomoleculares, os campos de força mais utilizados são CHARMM, GROMOS, AMBER, OPLS, entre outros.

# Dinâmica Molecular Clássica

Consiste da solução numérica, passo a passo, da Equação de movimento de Newton, que pode ser descrita para um sistema atômico simples como:

$$\mathbf{F}_i(t) = m_i \mathbf{a}_i$$



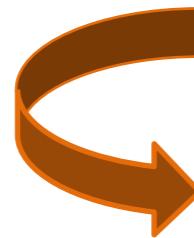
Integrando-se as equações do movimento:

Velocidades



Sua integral proporciona:

Mudança de posição do átomo



Com as novas posições e velocidades de cada partícula:

Energias Potencial e Cinética do sistema

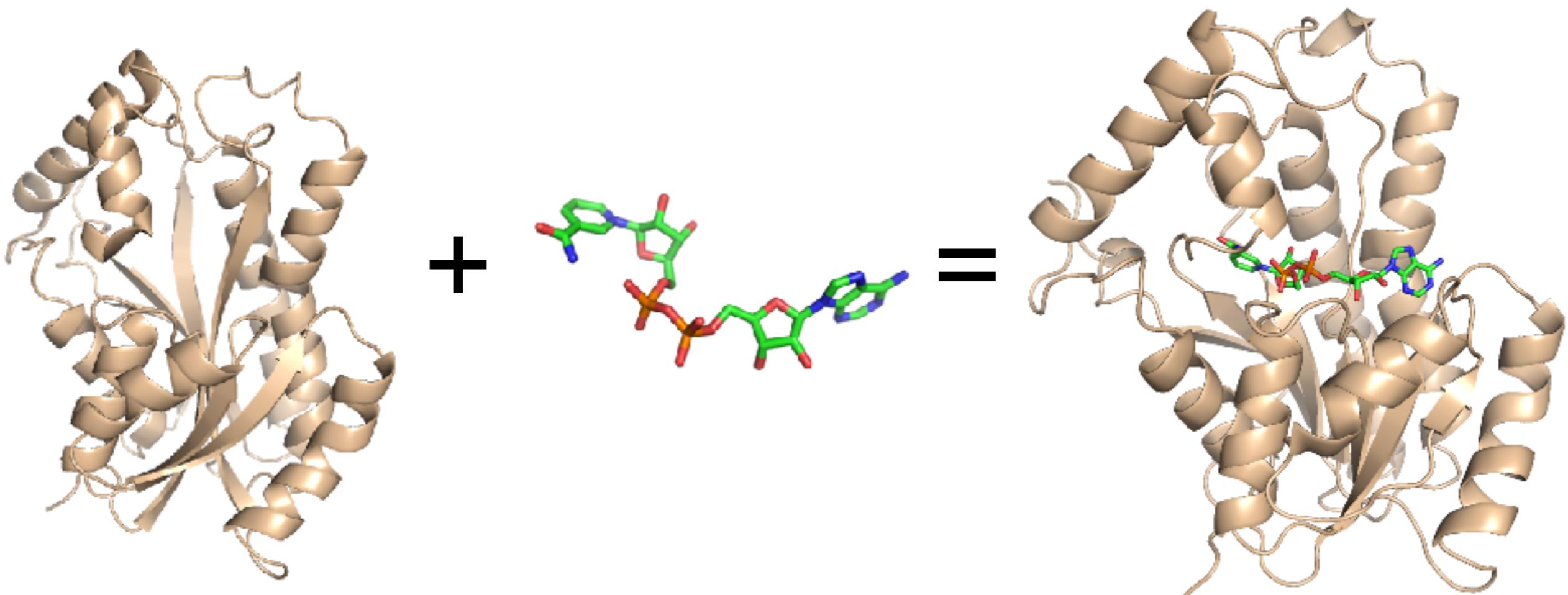


Aplicando-se sucessivamente este procedimento:

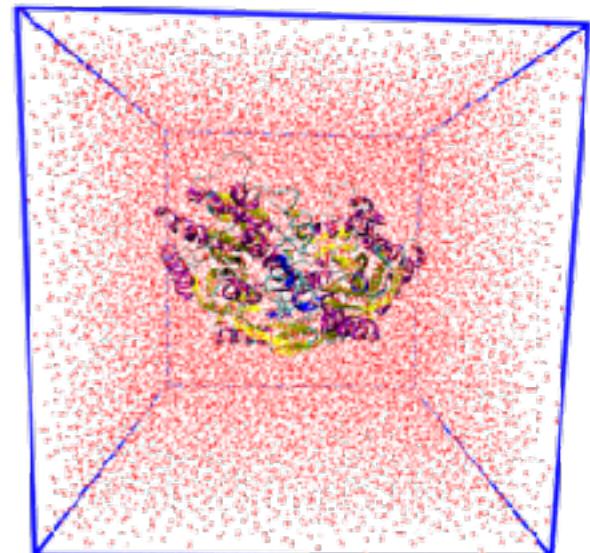
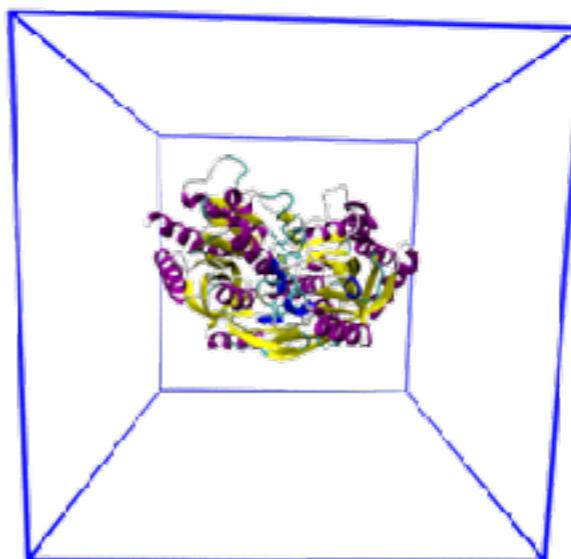
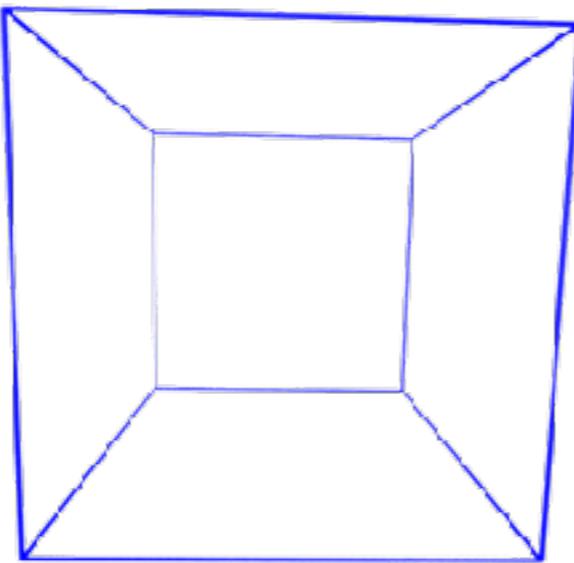
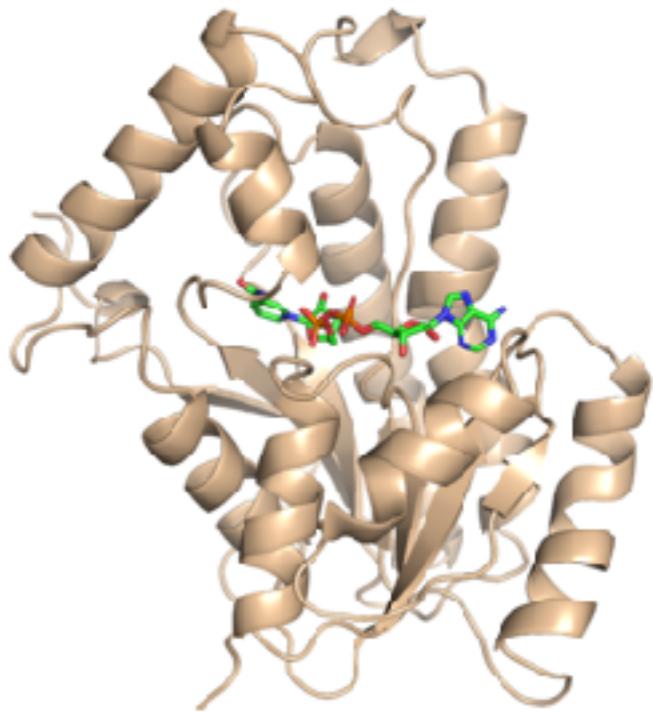
**TRAJETÓRIA** conj. posições e velocidades das partículas ao longo do tempo.

# Dinâmica Molecular Clássica

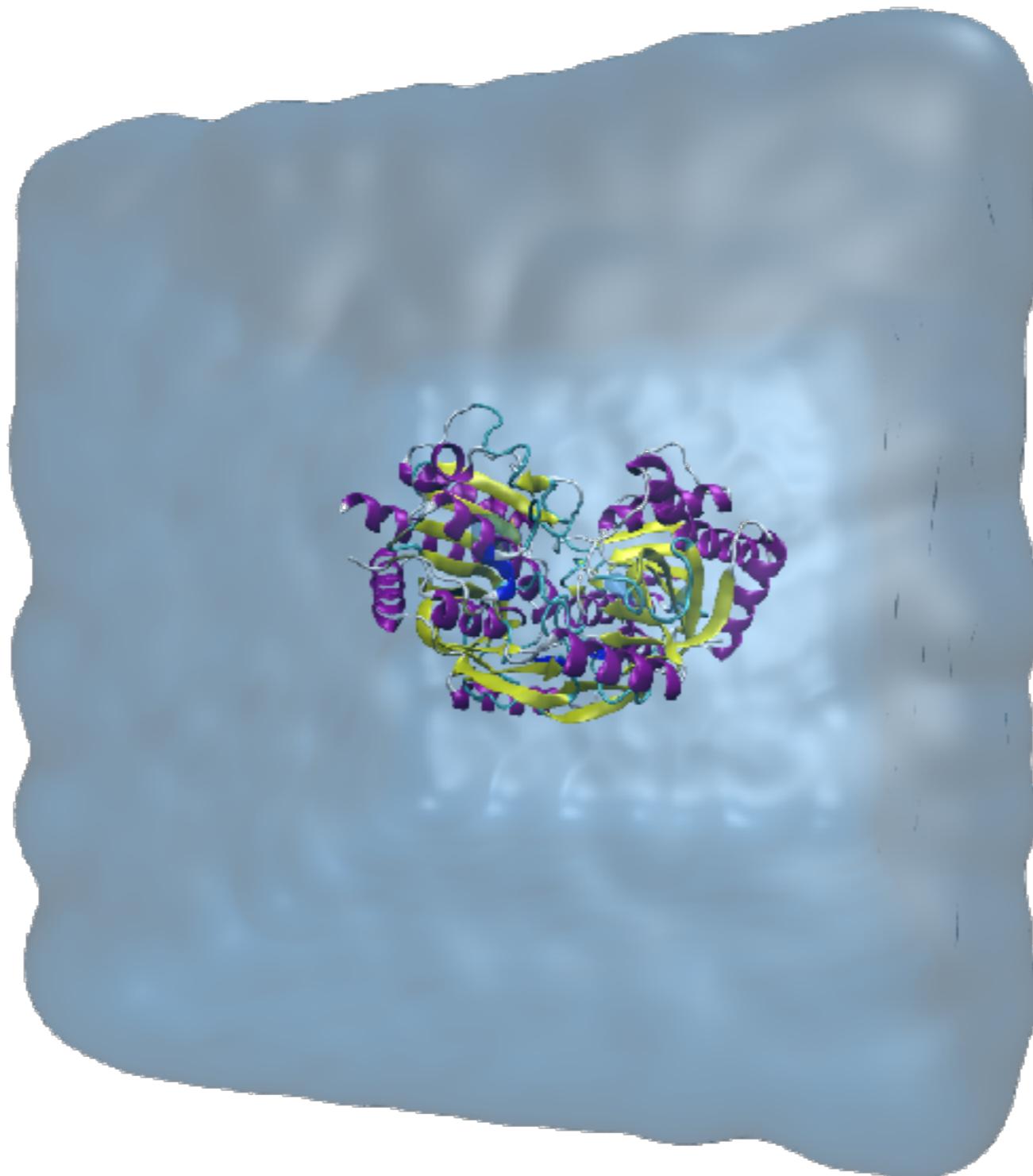
*Montagem de um sistema proteína-ligante*



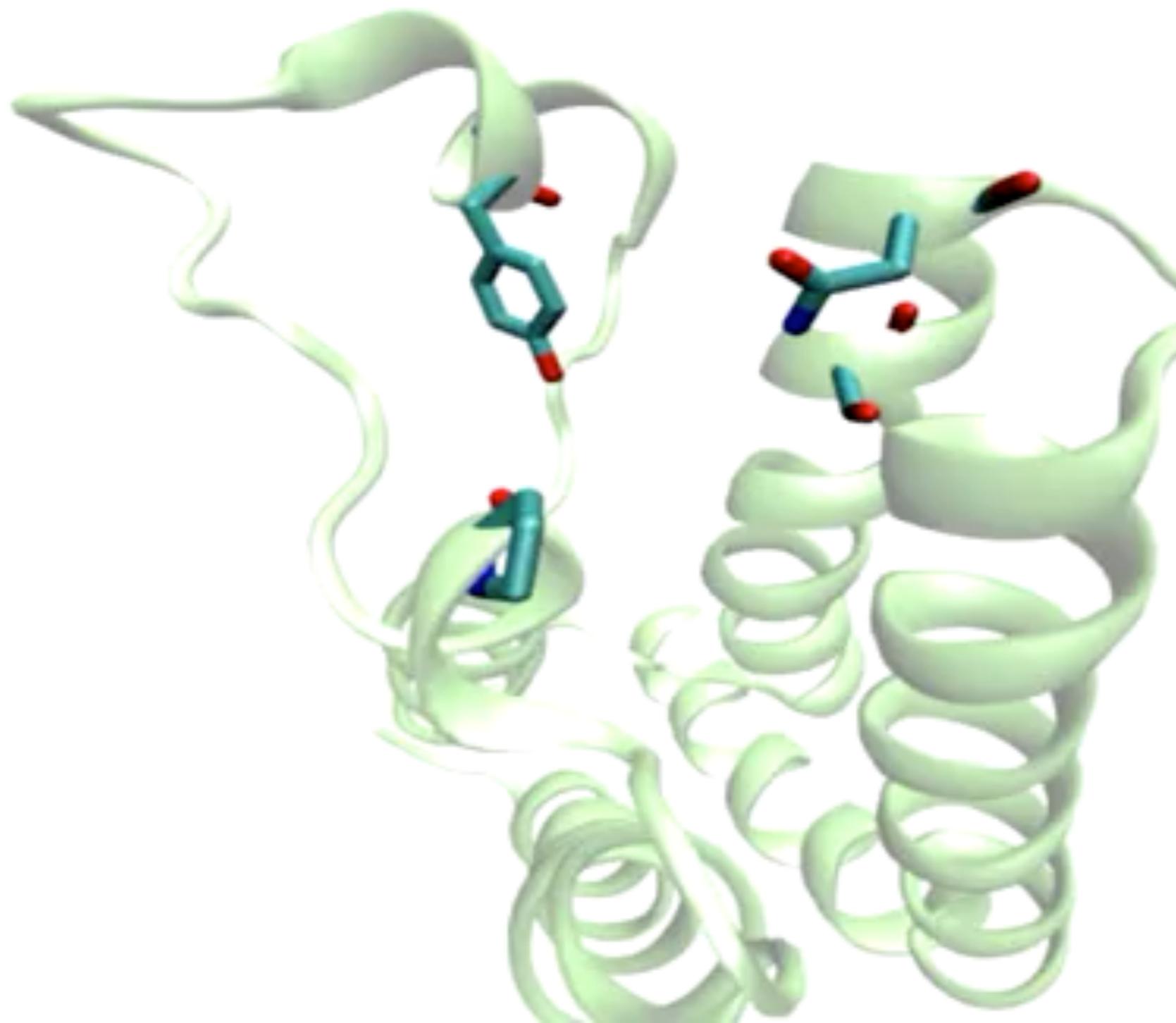
# Dinâmica Molecular Clássica



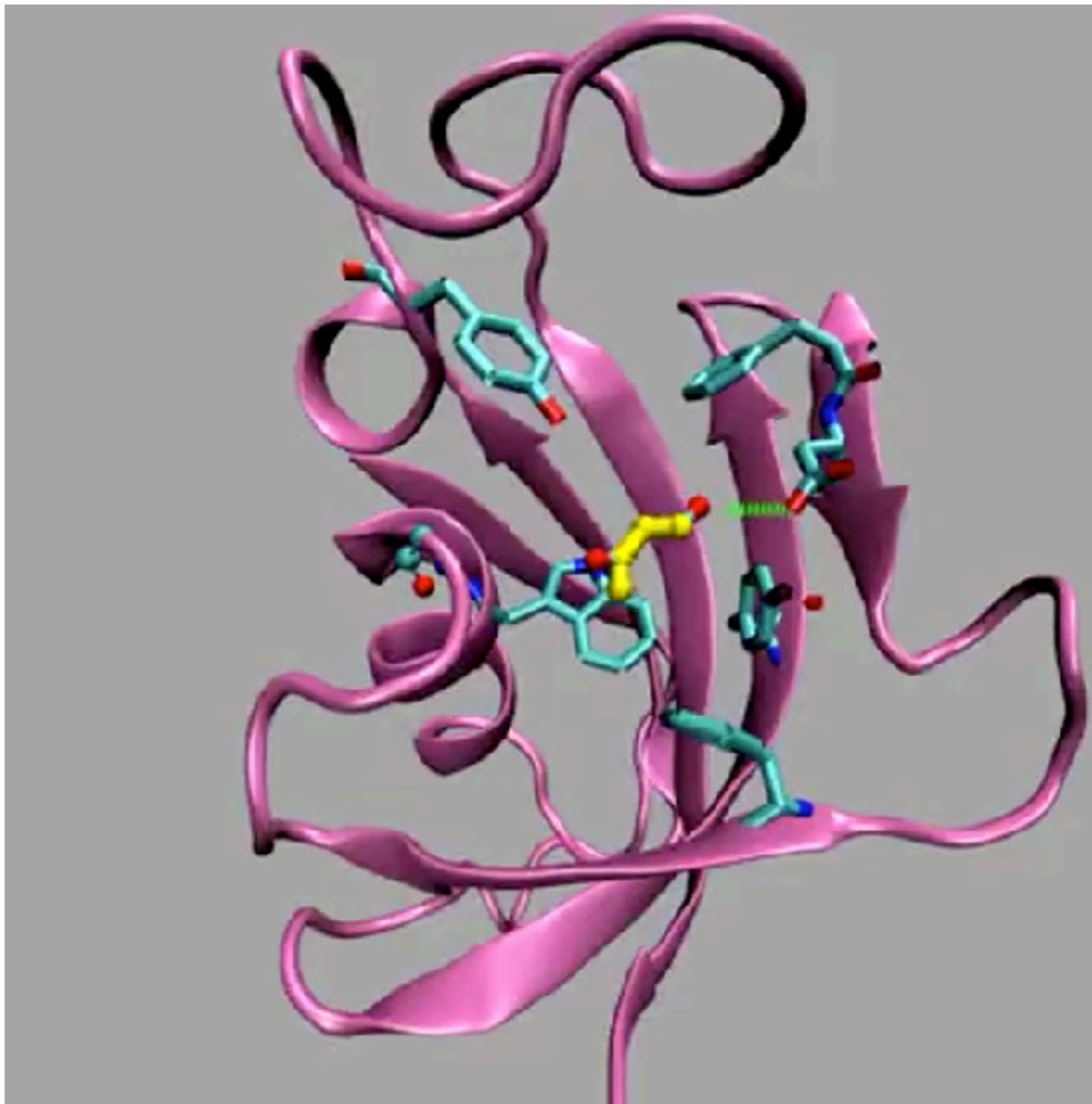
# Dinâmica Molecular Clássica



# Dinâmica Molecular Clássica



# Dinâmica Molecular Clássica



# DM e planejamento de fármacos

Simulações de DM têm sido rotineiramente utilizadas para investigar a estrutura e função de biomoléculas e seus respectivos complexos, e interações no processo de planejamento de fármacos assistido por computador.

- ✓ Incorpora **flexibilidade** de ambos, **ligante e receptor**, melhorando suas interações e reforçando a complementaridade entre eles;
- ✓ A habilidade de incorporar **moléculas de solvente** nas simulações de sistemas receptor-ligante é muito importante para o entendimento do **papel da água** e seus efeitos sobre a **estabilidade** de complexos proteína-inibidor.

# Preparação do Sistema

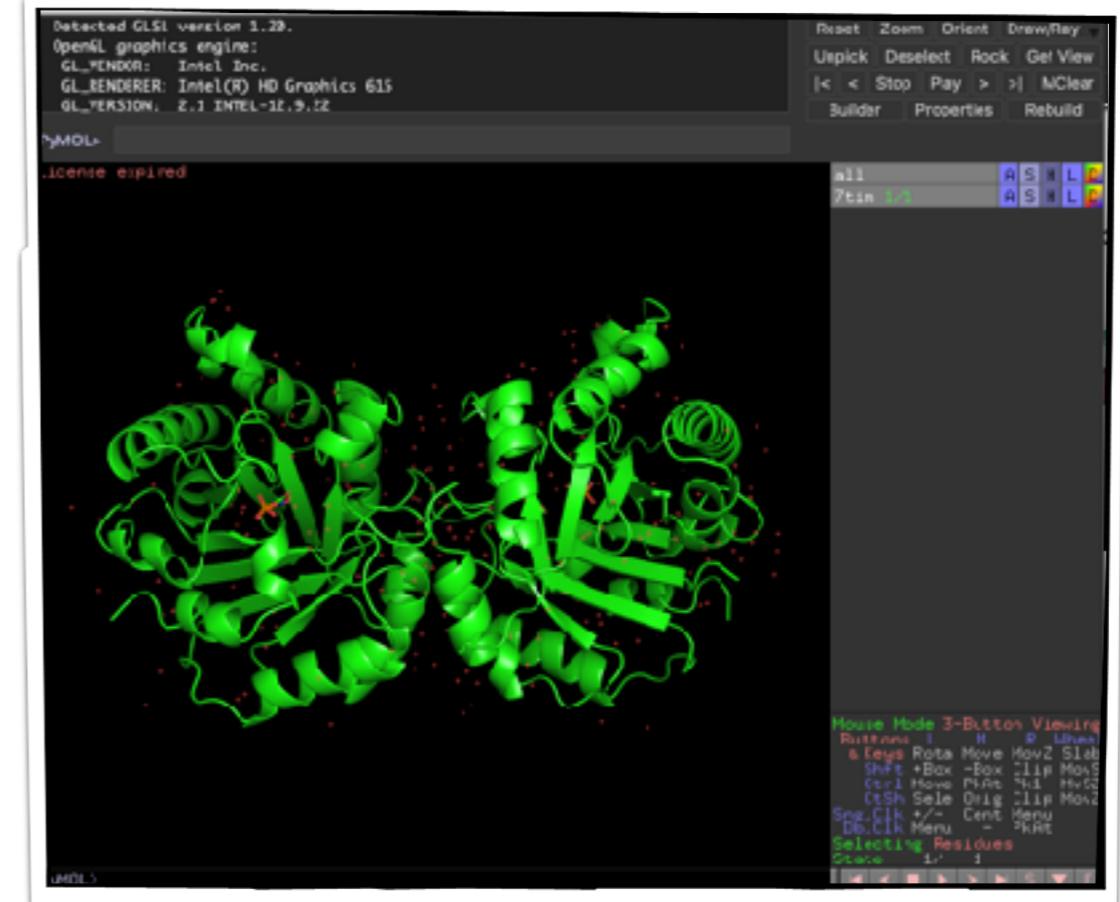
Para realizar uma simulação precisamos de um arquivo PDB da proteína de interesse

→ Neste tutorial utilizaremos a proteína Triosefósfato Isomerase (código PDB: **7TIM**)

*Editar o arquivo no PyMOL*

Acessar o site do PDB

STRUCTURE OF THE TRIOSEPHOSPHATE ISOMERASE-PHOSPHOGLYCEROLICACID COMPLEX AN ANALOGUE OF THE INTERMEDIATE ON THE REACTION PATHWAY  
PDB ID: 7TIM  
Experimental Data: X-ray Diffraction, Resolution: 1.8 Å  
Deposited By: RABINOWITZ J.D.  
Depositor Author(s): RABINOWITZ J.D., BAKER H.L., SIEBOLD H.A., KARRELL M., INGRAM G.P., HEDGES D.



Remover as H<sub>2</sub>O, deletar uma cadeia, assim como os ligantes

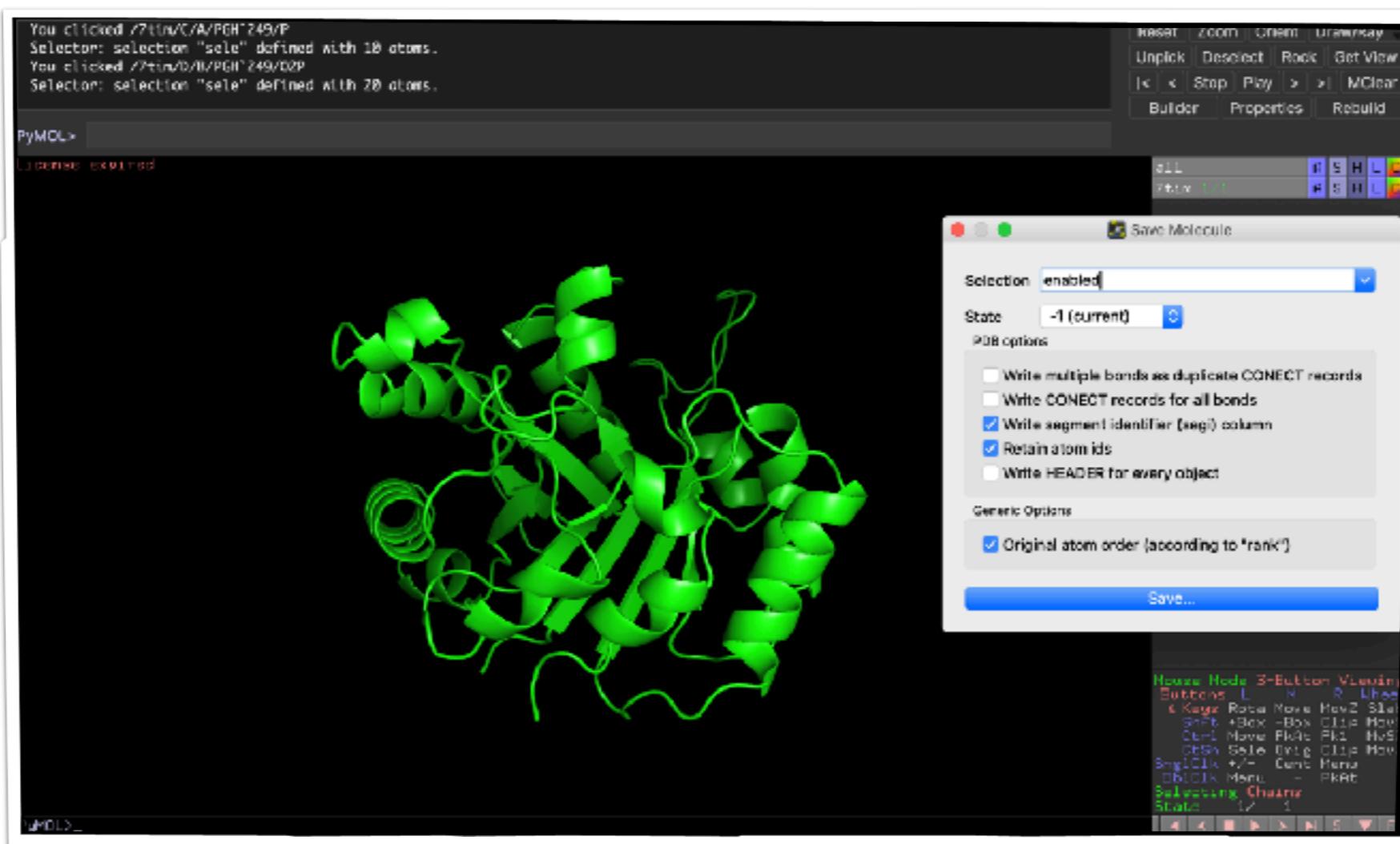
# Preparação do Sistema

Para realizar uma simulação precisamos de um arquivo PDB da proteína de interesse



Neste tutorial utilizaremos a proteína [Triosefósfato Isomerase](#) (código PDB: [7TIM](#))

*Editar o arquivo no PyMOL*

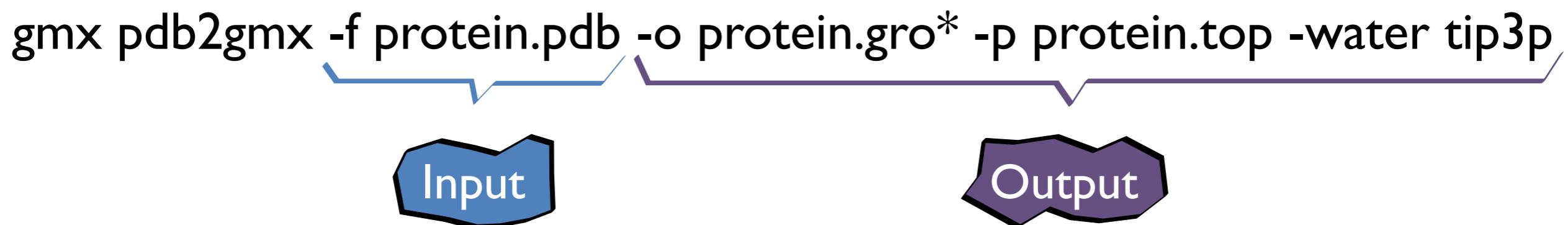


## Preparação do Sistema

A primeira etapa para a DM é a preparação do arquivo PDB conforme os parâmetros do campo de força selecionado.

→ A partir do arquivo PDB serão gerados **três** arquivos:

1. Arquivo de topologia (*.top*)
2. Arquivo de restrição de posições (*posre.itp*)
3. Arquivo de estrutura processado (*.gro*)



# Preparação do Sistema

A primeira etapa para a DM é a preparação do arquivo PDB conforme os parâmetros do campo de força selecionado.



No arquivo de topologia existem alguns parâmetros importantes:

```
; Include forcefield parameters  
#include "amber99sb.ff/forcefield.itp"
```

```
[ molecule type ]  
; Name nrexcl  
Protein_chain_B 3
```

Definição do campo de força

[ atoms ]											
;	nr	type	resnr	residue	atom	conc	charge	mass	typeB	chargeB	massB
;	residue	2	ALA	ALA	N	1	0.1414	14.01	;	atot	0.1414
	1	N3	1	ALA	H	2	0.1997	1.008	;	atot	0.3411
	2	H	2	ALA	H1	2	0.1997	1.008	;	atot	0.5408
	3	H	2	ALA	H2	3	0.1997	1.008	;	atot	0.7405
	4	H	2	ALA	H3	4	0.1997	1.008	;	atot	0.8367
	5	CT	2	ALA	CA	5	0.0962	12.01	;	atot	0.9256
	6	HP	2	ALA	HA	6	0.0889	1.008	;	atot	0.8659
	7	CT	2	ALA	CB	7	-0.0597	12.01	;	atot	0.8959
	8	HC	2	ALA	HB1	8	0.03	1.008	;	atot	0.9259
	9	HC	2	ALA	HB2	9	0.03	1.008	;	atot	0.9559
	10	HC	2	ALA	HB3	10	0.03	1.008	;	atot	1.572
	11	C	2	ALA	C	11	0.6163	12.01	;	atot	1
	12	O	2	ALA	O	12	-0.5722	16	;	atot	

→ Definição do nome da proteína

# Preparação do Sistema

A primeira etapa para a DM é a preparação do arquivo PDB conforme os parâmetros do campo de força selecionado.



No arquivo de topologia existem alguns parâmetros importantes:

```
; Include Position restraint file
#ifndef POSRES
#include "posre.itp"
#endif

; Include water topology
#include "amber99sb.ff/tip3p.itp"

#ifndef POSRES_WATER
; Position restraint for each water oxygen
[ position_restraints ]
; i funct      fcx      fcy      fcz
; 1   1        1000     1000     1000
#endif

; Include topology for ions
#include "amber99sb.ff/ions.itp"
```

Parâmetros para atribuir restrição de posição para alguns átomos da proteína

Parâmetros de topologia das moléculas de água

Parâmetros de topologia para os íons referente ao campo de força selecionado

# Preparação do Sistema

A primeira etapa para a DM é a preparação do arquivo PDB conforme os parâmetros do campo de força selecionado.



No arquivo de topologia existem alguns parâmetros importantes:  
...conferindo o arquivo de topologia...

```
[ system ]  
; Name  
Protein in water  
  
[ molecules ]  
; Compound      #mols  
Protein chain B    1
```

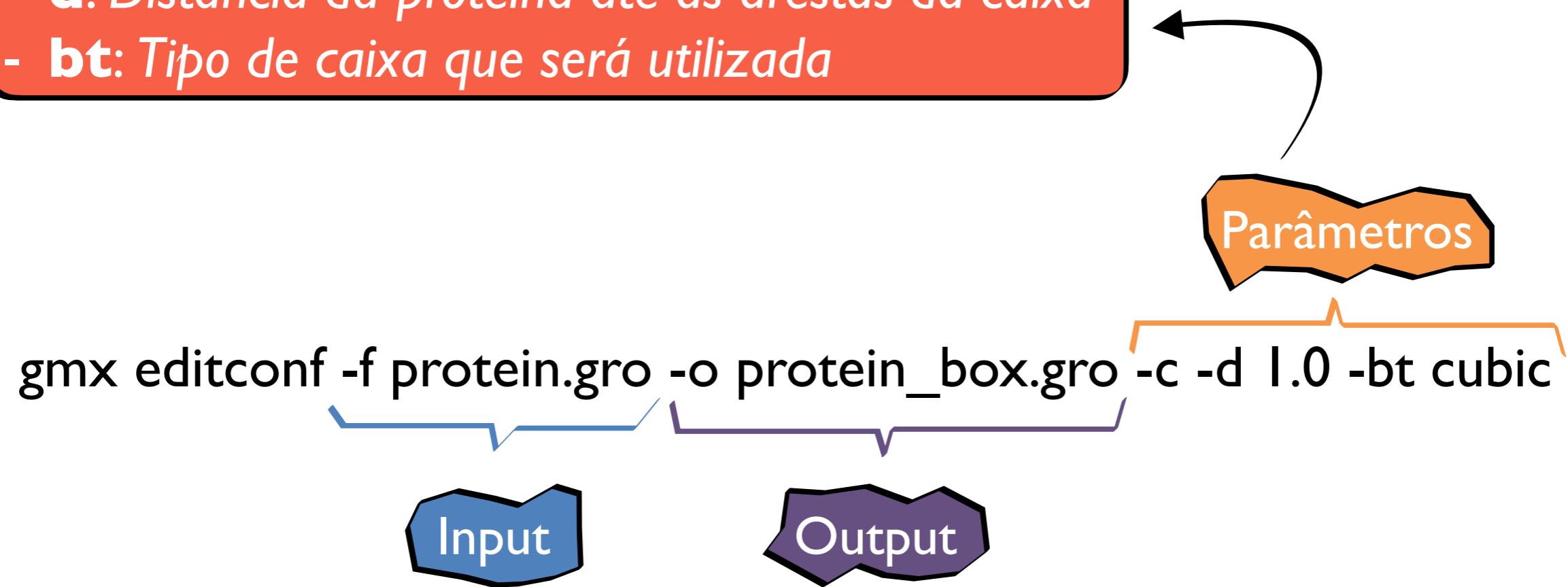
Resumo de todos os átomos  
presentes no sistema de  
estudo

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ (i) Definir as dimensões da caixa por meio do módulo “[editconf](#)”

- **c**: Centralizar a proteína no centro da caixa
- **d**: Distância da proteína até as arestas da caixa
- **bt**: Tipo de caixa que será utilizada



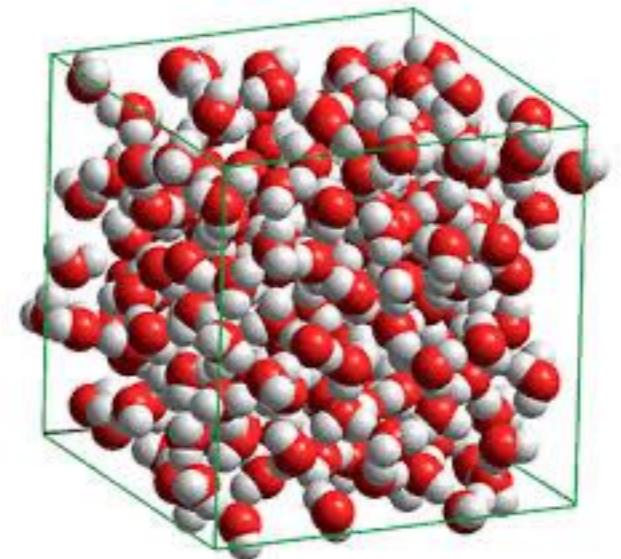
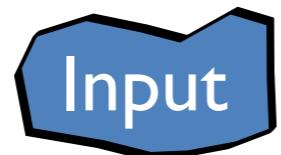
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

- (ii) Preencher a caixa gerada com as moléculas por meio do módulo “**solvate**”

- *Configuração dos parâmetros de solvente*

```
gmx solvate -cp protein_box.gro -p protein.top -cs -o protein_solv.gro
```



# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

- (ii) Preencher a caixa gerada com as moléculas por meio do módulo “**solvate**”

...conferindo o arquivo de topologia...

```
[ system ]  
; Name  
Protein in water  
  
[ molecules ]  
; Compound      #mols  
Protein chain B    1  
SOL                11595
```

Resumo de todos os átomos presentes no sistema de estudo

Incluindo as moléculas de  $H_2O$

Número de moléculas de  $H_2O$

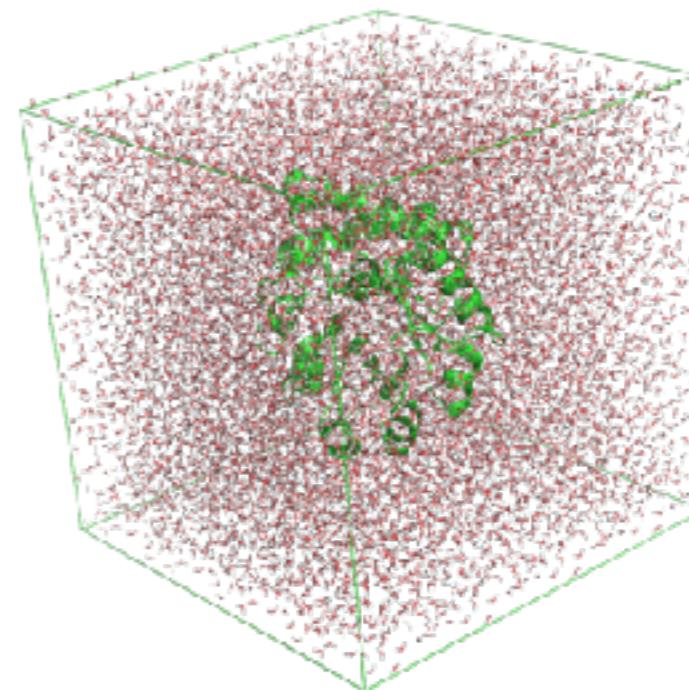
## Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

- (ii) Preencher a caixa gerada com as moléculas por meio do módulo “**solvate**”

...conferindo o [arquivo de simulação](#)...

Converte o arquivo “.gro” em  
“.pdb” para a visualização  
no PyMOL



`gmx editconf -f protein_solv.gro -o protein_box.pdb`

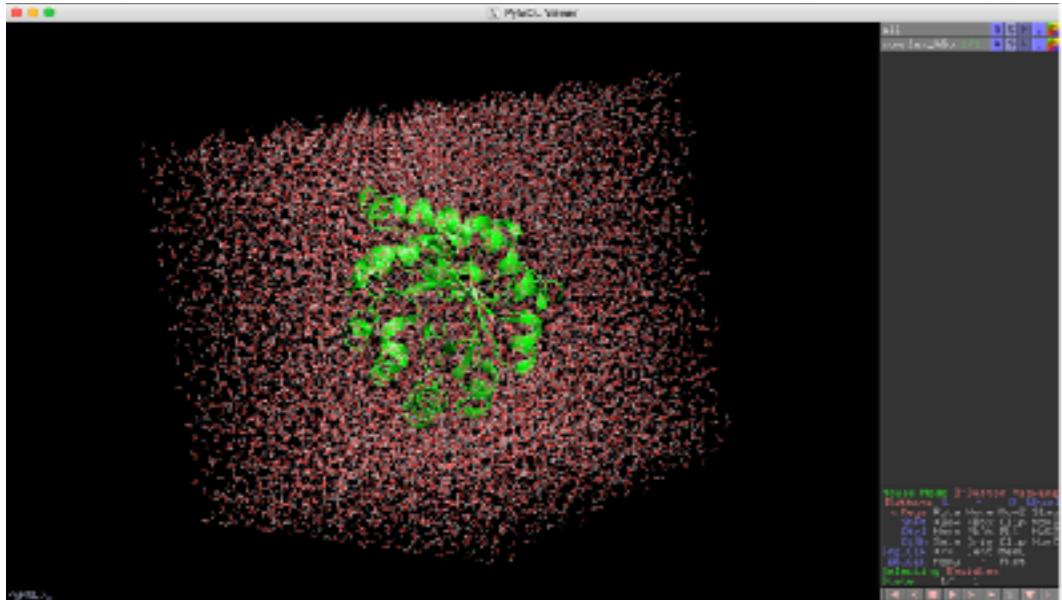
Input

Output

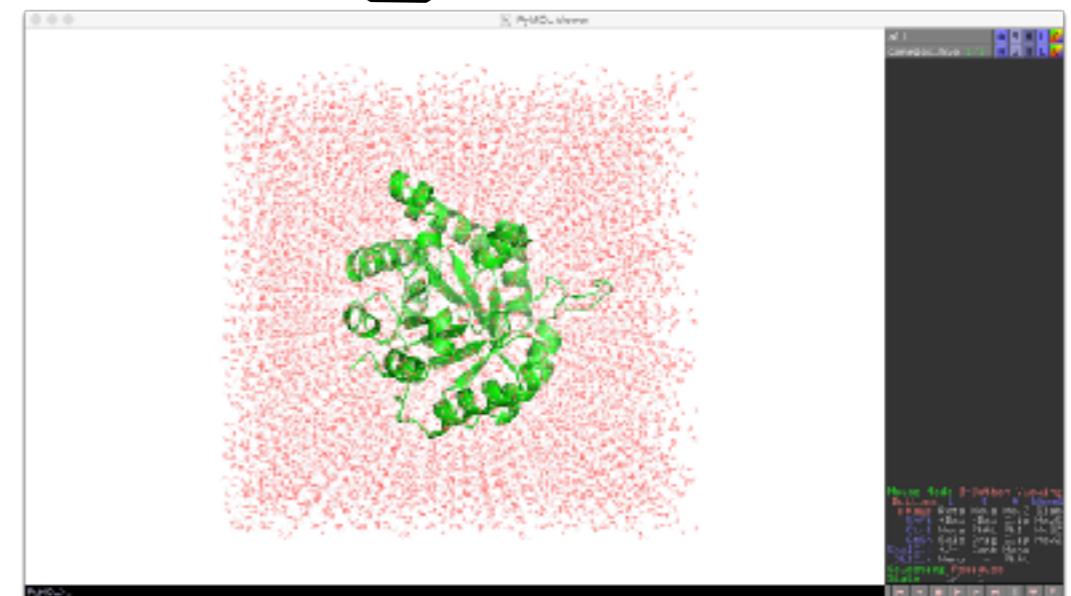
# Preparação do Sistema

Visualização do sistema no **PyMOL**

...digitar no terminal: **pymol protein\_box.pdb**

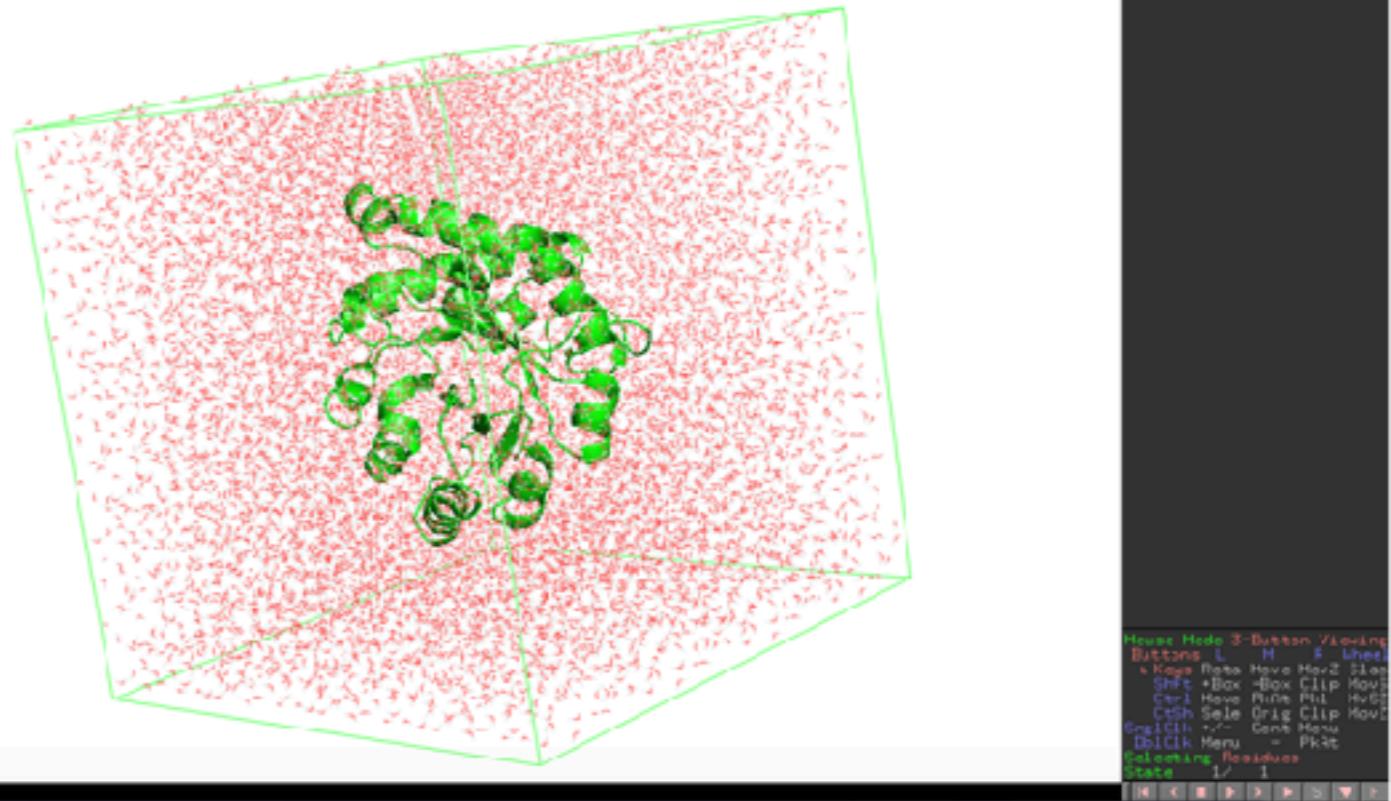


**>bg\_color white**



**>show cell**

Esse comando irá aparecer  
as delimitações da caixa  
construída para a  
simulação



# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Adicionar os íons com o intuito de neutralizar as cargas do sistema de estudo

```
gmx grompp -f ionsmdp -c protein_solv.gro -p protein.top -o ions.tpr
```

Input

Output

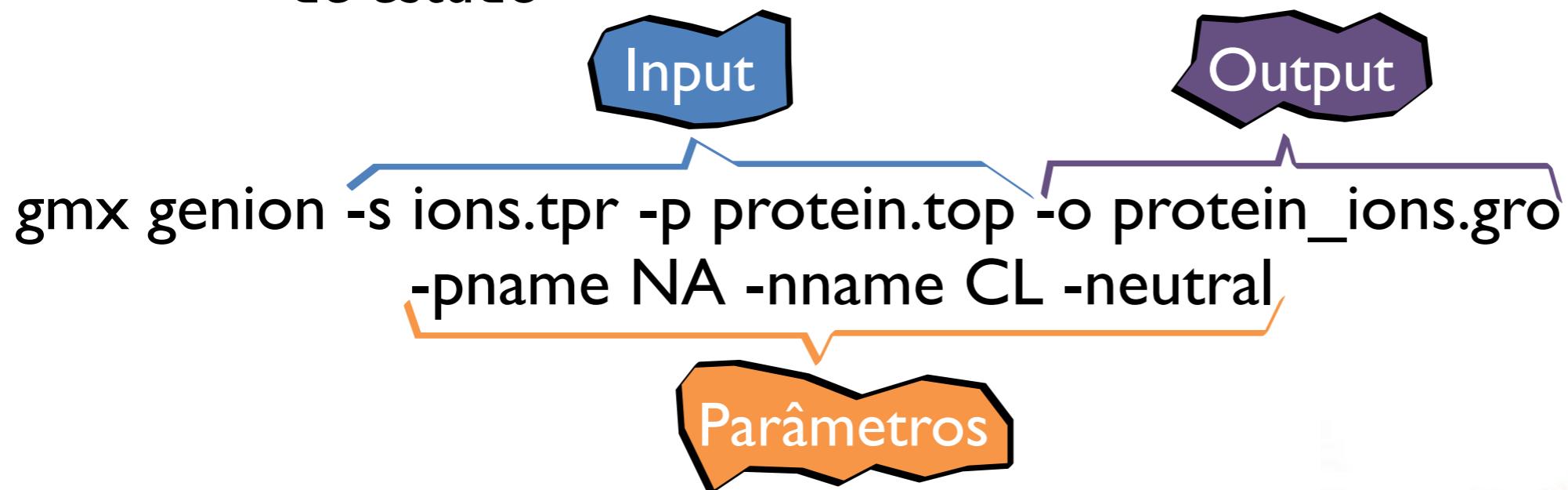
```
NOTE 2 [file protein_2.top, line 35614]:  
System has non-zero total charge: -5.000002  
Total charge should normally be an integer. See  
http://www.gromacs.org/Documentation/Floating\_Point\_Arithmetic  
for discussion on how close it should be to an integer.
```

Descreve a carga do sistema,  
informando quantos íons  
deverão ser adicionados

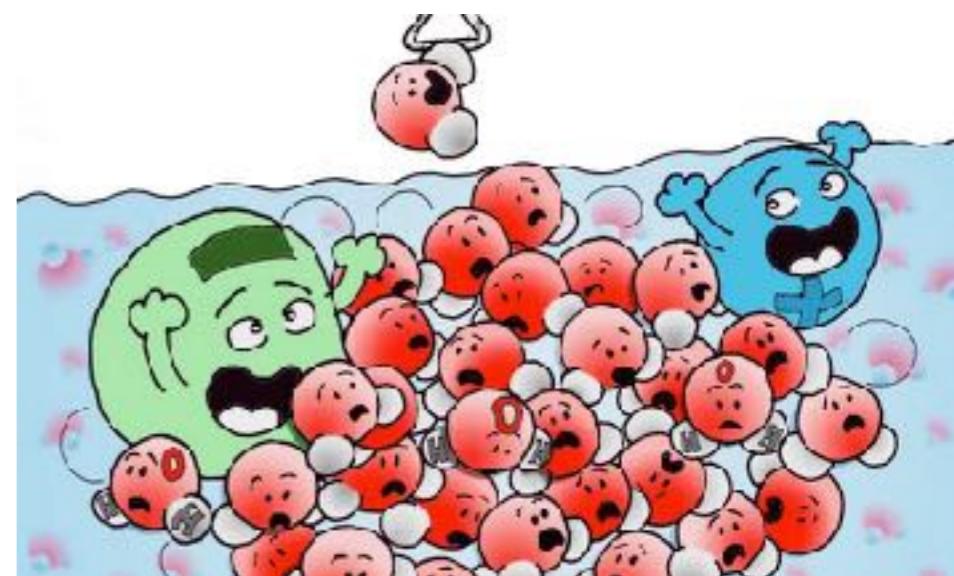
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Adicionar os íons com o intuito de neutralizar as cargas do sistema de estudo



Determina que serão adicionados o número exato de íons para neutralizar o sistema. Existe a possibilidade de adicionar os íons de acordo com uma concentração específica, i.e. “**-conc 0.15 -neutral**”



# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM



Adicionar os ions com o intuito de neutralizar as cargas do sistema de estudo

```
Reading file ions.tpr, VERSION 2018.1 (single precision)
Reading file ions.tpr, VERSION 2018.1 (single precision)
Will try to add 5 NA ions and 0 CL ions.
Select a continuous group of solvent molecules
Group      0 (      System) has 38578 elements
Group      1 (      Protein) has 3778 elements
Group      2 (  Protein-H) has 1883 elements
Group      3 (    C-alpha) has   247 elements
Group      4 (    Backbone) has   741 elements
Group      5 ( MainChain) has   989 elements
Group      6 ( MainChain+Cb) has 1214 elements
Group      7 ( MainChain+H) has 1231 elements
Group      8 ( SideChain) has 2547 elements
Group      9 ( SideChain-H) has   894 elements
Group     10 ( Prot-Masses) has 3778 elements
Group     11 ( non-Protein) has 34800 elements
Group     12 (      Other) has    15 elements
Group     13 (        PGH) has    15 elements
Group     14 (       Water) has 34785 elements
Group     15 (        SOL) has 34785 elements
Group     16 ( non-Water) has  3793 elements
Select a group: 15
```

**Selecionar o grupo que esteja o solvete! Serão trocadas moléculas de água por ions!!!**

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Adicionar os íons com o intuito de neutralizar as cargas do sistema de estudo

...conferindo o arquivo de topologia...

```
[ molecules ]
; Compound          #mols
Protein_chain_B    1|
SOL                11590
NA                 5
```

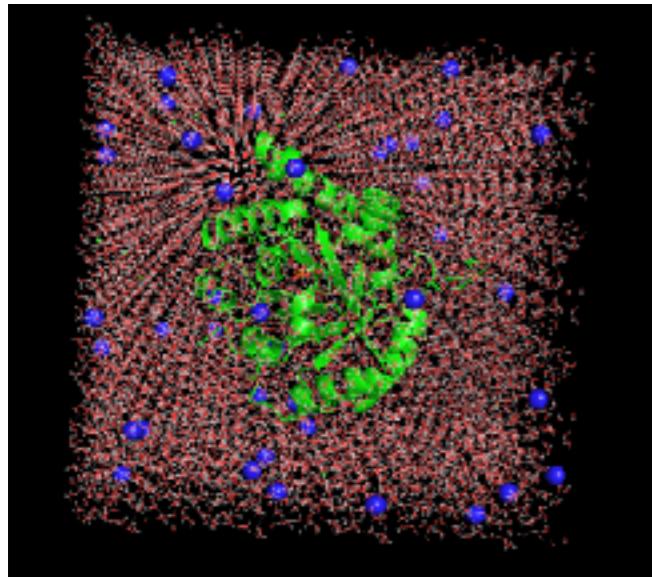
Resumo de todos os átomos  
presentes no sistema de  
estudo

Foram trocadas **cinco** moléculas de **H<sub>2</sub>O**  
por **cinco** íons de **Na<sup>+</sup>**

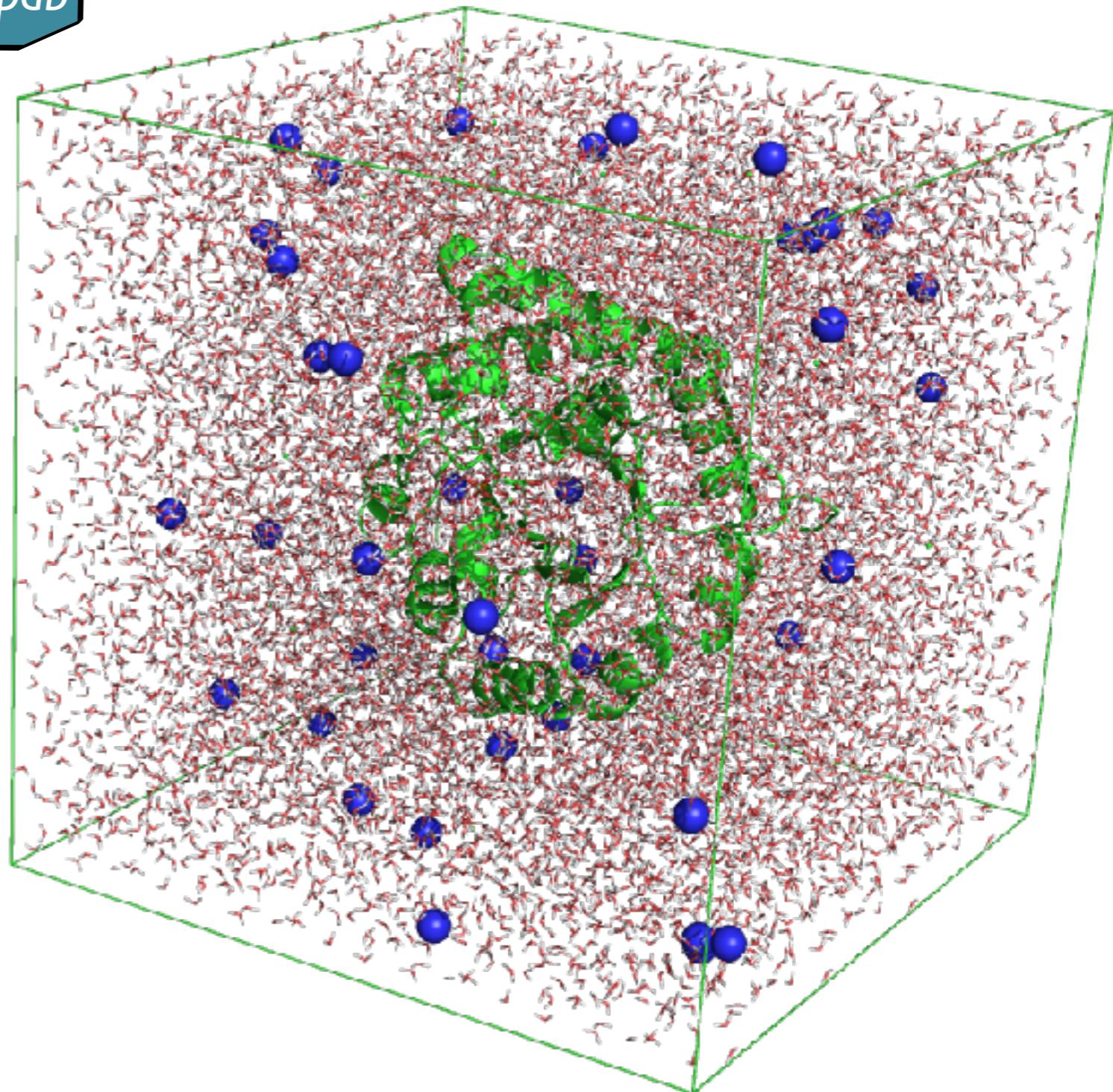
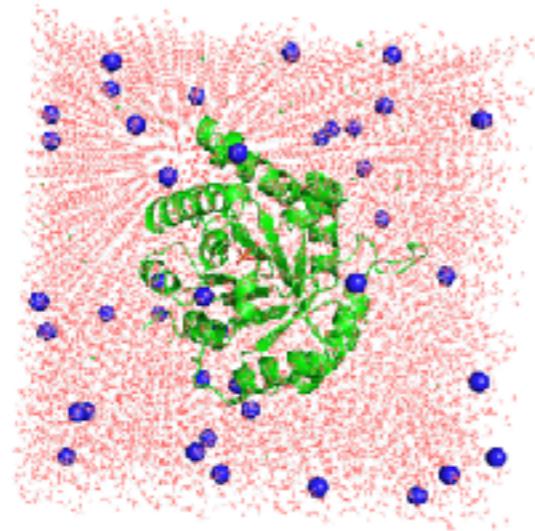
# Preparação do Sistema

Visualização do sistema no **PyMOL**

...digitar no terminal: **pymol protein\_ions.pdb**



**>bg\_color white**

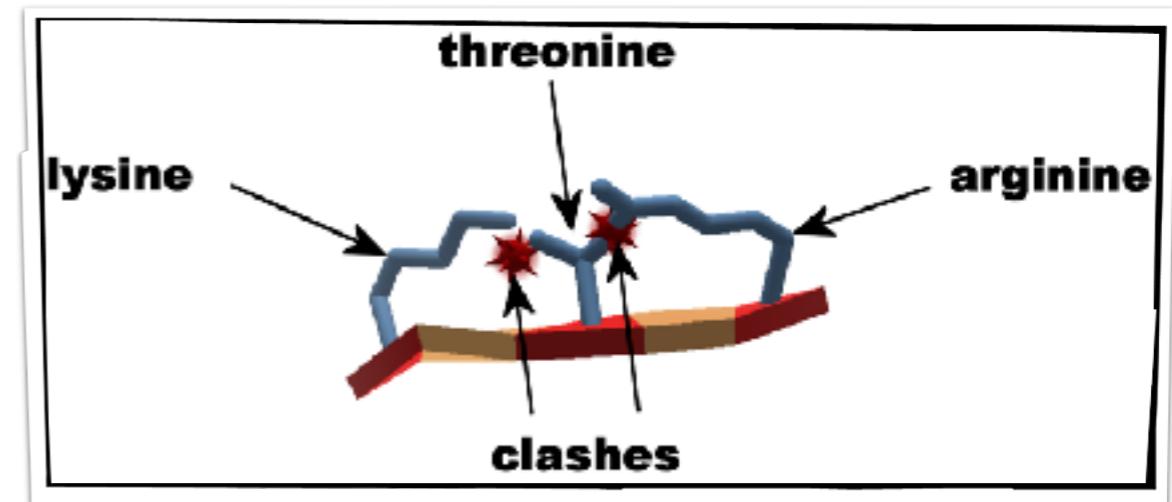


# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Minimização do sistema para evitar geometrias incorretas, assim como, choques entre cadeias laterais do resíduos

Etapa de pré-  
processamento



gmx grompp -f minim.mdp -c protein\_ions.gro -p protein.top -o em.tpr

Input

Output

This run will generate roughly 3 Mb of data

Fornece uma estimativa de espaço!

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Minimização do sistema para evitar geometrias incorretas, assim como, choques entre cadeias laterais do resíduos

Esta etapa irá gerar quatro arquivos:

- (i) em.log: arquivo de texto do log do processo de minimização
- (ii) em.edr: arquivo binário de energia
- (iii) em.trr: arquivo binário da trajetória
- (iv) em.gro: Estrutura final minimizada

...opção para impacientes!

gmx mdrun -deffnm em

-v

Input

```
Step= 481, Dmax= 9.7e-03 nm, Epot= -5.70880e+05 Fmax= 3.86465e+03, atom= 519
Step= 483, Dmax= 5.8e-03 nm, Epot= -5.70908e+05 Fmax= 5.01022e+03, atom= 519
Step= 484, Dmax= 7.0e-03 nm, Epot= -5.70936e+05 Fmax= 5.72636e+03, atom= 519
Step= 485, Dmax= 8.4e-03 nm, Epot= -5.70953e+05 Fmax= 7.06862e+03, atom= 519
Step= 486, Dmax= 1.0e-02 nm, Epot= -5.70966e+05 Fmax= 8.36949e+03, atom= 519
Step= 488, Dmax= 6.0e-03 nm, Epot= -5.71068e+05 Fmax= 8.50719e+02, atom= 519

writing lowest energy coordinates.

Steepest Descents converged to Fmax < 1000 in 489 steps
Potential Energy      = -5.7106756e+05
Maximum force          =  8.5071869e+02 on atom 519
Norm of force           =  2.7358357e+01
```

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Minimização do sistema para evitar geometrias incorretas, assim como, choques entre cadeias laterais do resíduos

Gráfico no **Xmgrace**

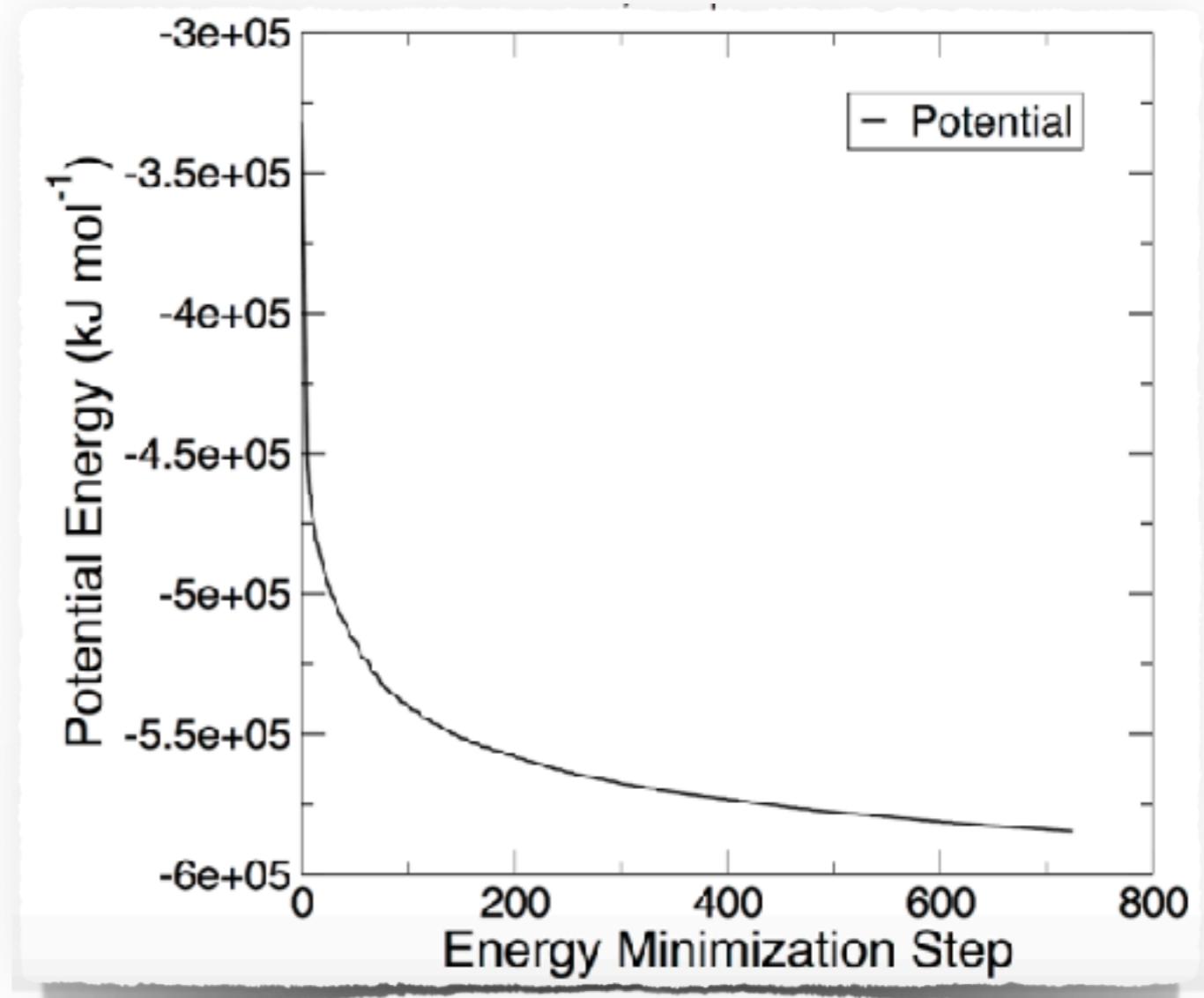
A partir do arquivo de energia obtido (.edr) criar um arquivo para plotar os dados

```
gmx energy -f em.edr -o potential.xvg
```

Input      Output

I O      O

xmgrace potential.xvg



# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Equilibrar o as moléculas do solvente ( $H_2O$ ) ao redor da proteína, permitindo uma melhor acomodação das  $H_2O$  nas cavidades da molécula de estudo

Para garantir que apenas as moléculas de  $H_2O$  se movam, utilizaremos aqui o arquivo “posre.itp”

Aplica uma força sobre os átomos pesados da proteína

```
[ position_restraints ]
; atom   type    fx      fy      fz
    1      1    1000    1000    1000
    5      1    1000    1000    1000
    7      1    1000    1000    1000
   11      1    1000    1000    1000
   12      1    1000    1000    1000
   13      1    1000    1000    1000
   15      1    1000    1000    1000
   17      1    1000    1000    1000
   20      1    1000    1000    1000
   23      1    1000    1000    1000
   26      1    1000    1000    1000
   28      1    1000    1000    1000
   29      1    1000    1000    1000
```

Geração de velocidades para uma temperatura específica

```
; Velocity generation
gen_vel          = yes
gen_temp         = 300
gen_seed         = -1
```

```
define           = -DPOSRES ; position restrain the protein
; Run parameters
integrator      = md        ; leap-frog integrator
nsteps          = 50000    ; 2 * 50000 = 100 ps
dt              = 0.002    ; 2 fs
```

gmx grompp -f nvtmdp -c em.gro -r em.gro -p topol.top -o nvt.tpr

Input

Output

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM



Equilibrar o as moléculas do solvente ( $H_2O$ ) ao redor da proteína, permitindo uma melhor acomodação das  $H_2O$  nas cavidades da molécula de estudo

Esta etapa irá gerar quatro arquivos:

- (i) nvt.log: arquivo de texto do log do processo de minimização
- (ii) nvt.edr: arquivo binário de energia
- (iii) nvt.trr: arquivo binário da trajetória
- (iv) nvt.gro: Estrutura final minimizada

...opção para impacientes!

gmx mdrun -deffnm nvt

-v

Input

O comando “**mdrun**” irá executar a simulação e incluindo a opção “**-v**” teremos uma estimativa de quanto tempo irá levar

```
Using 4 MPI threads
Back Off! I just backed up nvt.trr to ./#nvt.trr.1#
Back Off! I just backed up nvt.edr to ./#nvt.edr.1#
starting mdrun 'Protein in water'
50000 steps, 100.0 ps.
step 0imb F 6%
step 100 Turning on dynamic load balancing, because the performance
step 4400, will finish Mon Jul 22 09:58:26 2019 Vol 0.93 imb F 1%
```

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Uma vez equilibrada a temperatura, precisamos equilibrar a pressão do sistema

Para garantir que apenas as moléculas de  $H_2O$  se movam, utilizaremos aqui o arquivo “posre.itp”

Aplica uma força sobre os átomos pesados da proteína

```
[ position_restraints ]
; atom   type      fx      fy      fz
    1      1  1000  1000  1000
    5      1  1000  1000  1000
    7      1  1000  1000  1000
   11      1  1000  1000  1000
   12      1  1000  1000  1000
   13      1  1000  1000  1000
   15      1  1000  1000  1000
   17      1  1000  1000  1000
   20      1  1000  1000  1000
   23      1  1000  1000  1000
   26      1  1000  1000  1000
   28      1  1000  1000  1000
   29      1  1000  1000  1000
```

Como utilizaremos a mesma temperatura do passo anterior não é necessário gerar velocidades

```
; Velocity generation
gen_vel = no
```

E utilizamos uma opção para continuar a partir da etapa anterior

```
continuation = yes ; Restarting after NVT
define          = -DPOSRES ; position restrain the protein
; Run parameters
integrator     = md       ; leap-frog integrator
nsteps          = 50000   ; 2 * 50000 = 100 ps
dt              = 0.002   ; 2 fs
```

gmx grompp -f nptmdp -c nvt.gro -r nvt.gro -t nvt.cpt -p protein.top -o npt.tpr

Input

Output

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ *Uma vez equilibrada a temperatura, precisamos equilibrar a pressão do sistema*

Esta etapa irá gerar quatro arquivos:

- (i) *npt.log*: arquivo de texto do log do processo de minimização
- (ii) *npt.edr*: arquivo binário de energia
- (iii) *npt.trr*: arquivo binário da trajetória
- (iv) *npt.gro*: Estrutura final minimizada

...opção para impacientes!

gmx mdrun -deffnm npt

-v

Input

O comando “***mdrun***” irá executar a simulação e incluindo a opção “***-v***” teremos uma estimativa de quanto tempo irá levar

```
Using 4 MPI threads
starting mdrun 'Protein in water'
50000 steps, 100.0 ps.
step 100, will finish Mon Jul 22 10:48:56 2019
step 150 Turning on dynamic load balancing, because the performance
step 1000, will finish Mon Jul 22 10:46:49 2019 01 0.93 imb F 2%
```

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM



*Uma vez equilibrada a temperatura, precisamos equilibrar a pressão do sistema*

## Gráfico no Xmgrace

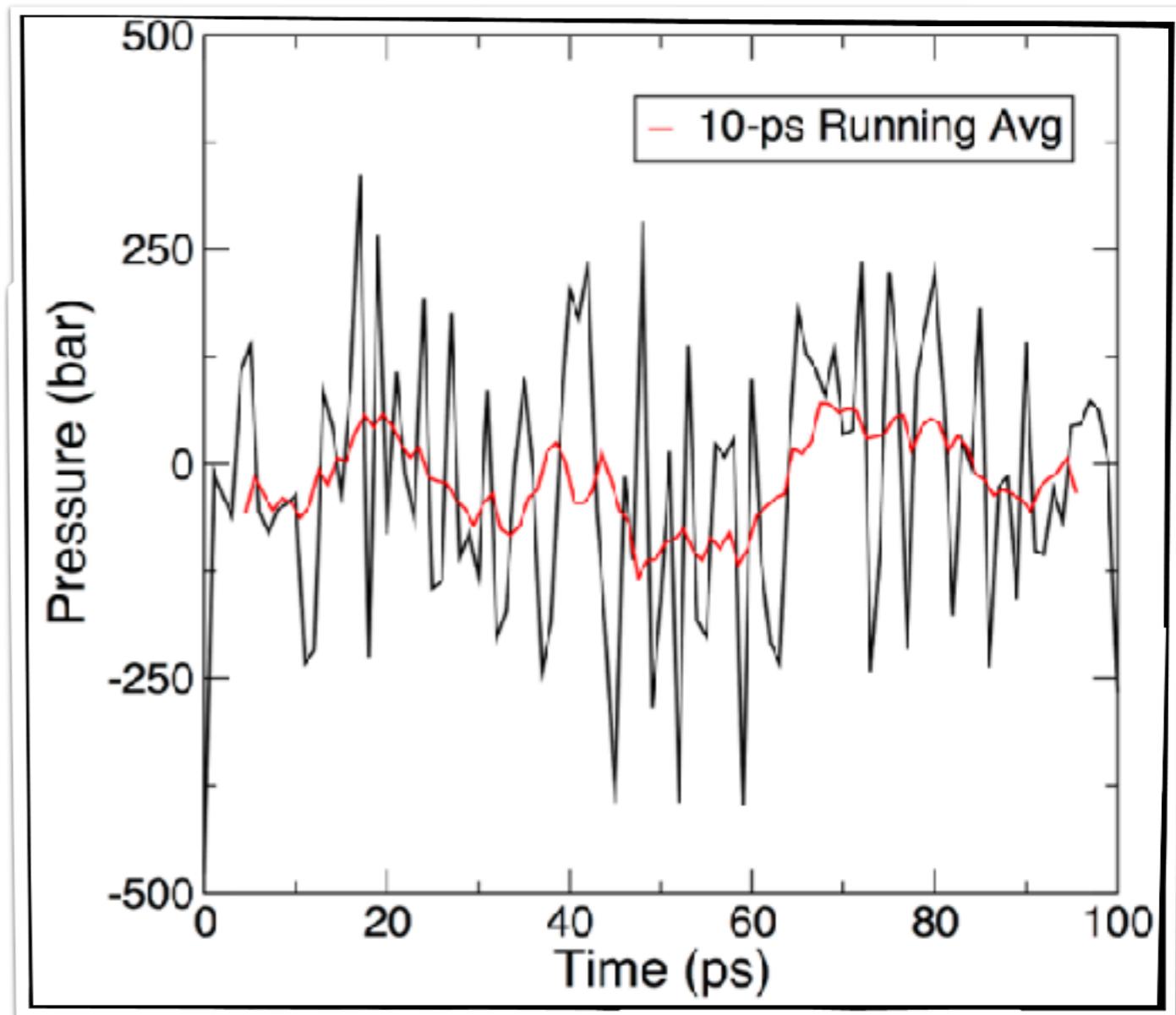
A partir do arquivo de energia obtido (.edr) criar um arquivo para plotar os dados

`gmx energy -f npt.edr -o pressure.xvg`



**I8**      **0**

`xmgrace pressure.xvg`



# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ *Com o sistema devidamente equilibrado, podemos iniciar com a fase de produção do dinâmica molecular!*

Nesta etapa não são mais aplicadas restrições a nenhum tipo de átomo presente no sistema

Todos os átomos estão livres para se movimentar

*Não é mais aplicada a opção “-DPOSRE” e sim um integrador para a DM “md”*

```
; Run parameters
integrator          = md           ; leap-frog integrator
nsteps              = 500000       ; 2 * 250000 = 500 ps (1 ns)
dt                  = 0.002        ; 2 fs
```

Continuamos sem gerar novas velocidades

```
; Velocity generation
gen_vel             = no
```

gmx grompp -f md.mdp -c npt.gro -t npt cpt -p protein.top -o md.tpr

Input

Output

# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ *Com o sistema devidamente equilibrado, podemos iniciar com a fase de produção do dinâmica molecular!*

Esta etapa irá gerar quatro arquivos:

- (i) *md.log: arquivo de texto do log do processo de minimização*
- (ii) *md.edr: arquivo binário de energia*
- (iii) *md.xtc: arquivo binário da trajetória*
- (iv) *md.gro: Estrutura final minimizada*

...opção para impacientes!

gmx mdrun -deffnm md



Input

O comando “**mdrun**” irá executar a simulação e incluindo a opção “**-v**” teremos uma estimativa de quanto tempo irá levar

```
Using 4 MPI threads

Back Off! I just backed up md.xtc to ./#md.xtc.1#
Back Off! I just backed up md.edr to ./#md.edr.1#
starting mdrun 'Protein in water'
500000 steps, 1000.0 ps.
step 100, will finish Mon Jul 22 14:44:24 2019
step 150 Turning on dynamic load balancing, because the performance
step 1000, will finish Mon Jul 22 14:15:38 2019@1 0.94 imb F 2%
```

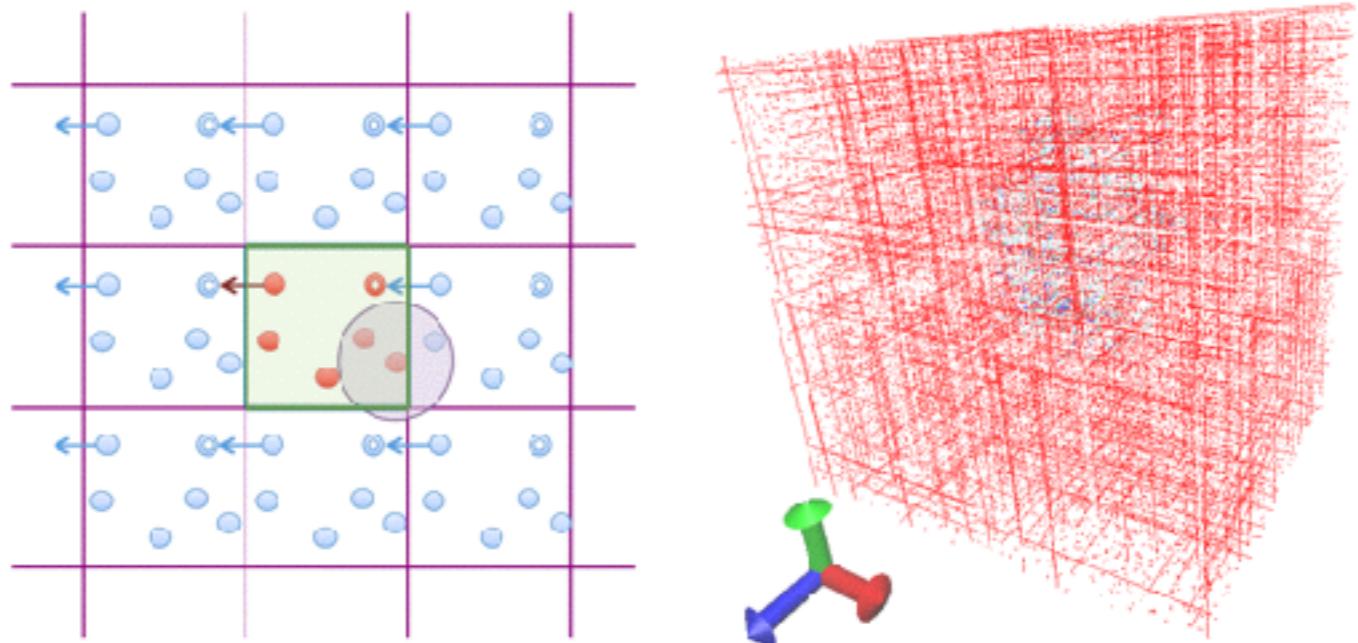
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

Condições periódicas de contorno (PBC)

A partir deste ponto iremos centralizar todos os frames da proteína no centro da caixa do sistema



```
gmx trjconv -s md.tpr -f md.xtc -pbc mol -center -o md_noPBC.xtc
```

Input

Parâmetros

Output

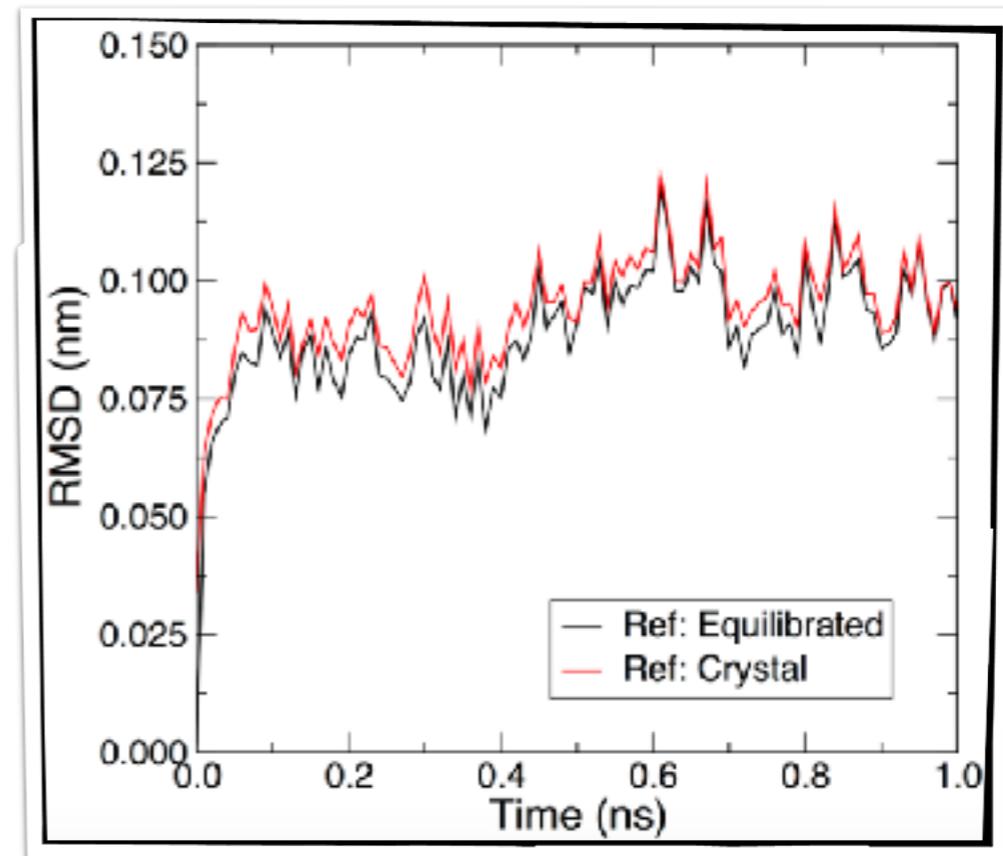
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

## Desvio médio quadrático (RMSD)

Descreve a variação da estrutura da proteína ao longo da simulação em relação a estrutura inicial



```
gmx rms -s md.tpr -f md_noPBC.xtc -o rmsd.xvg -tu ns
```

Input

Output

Parâmetros

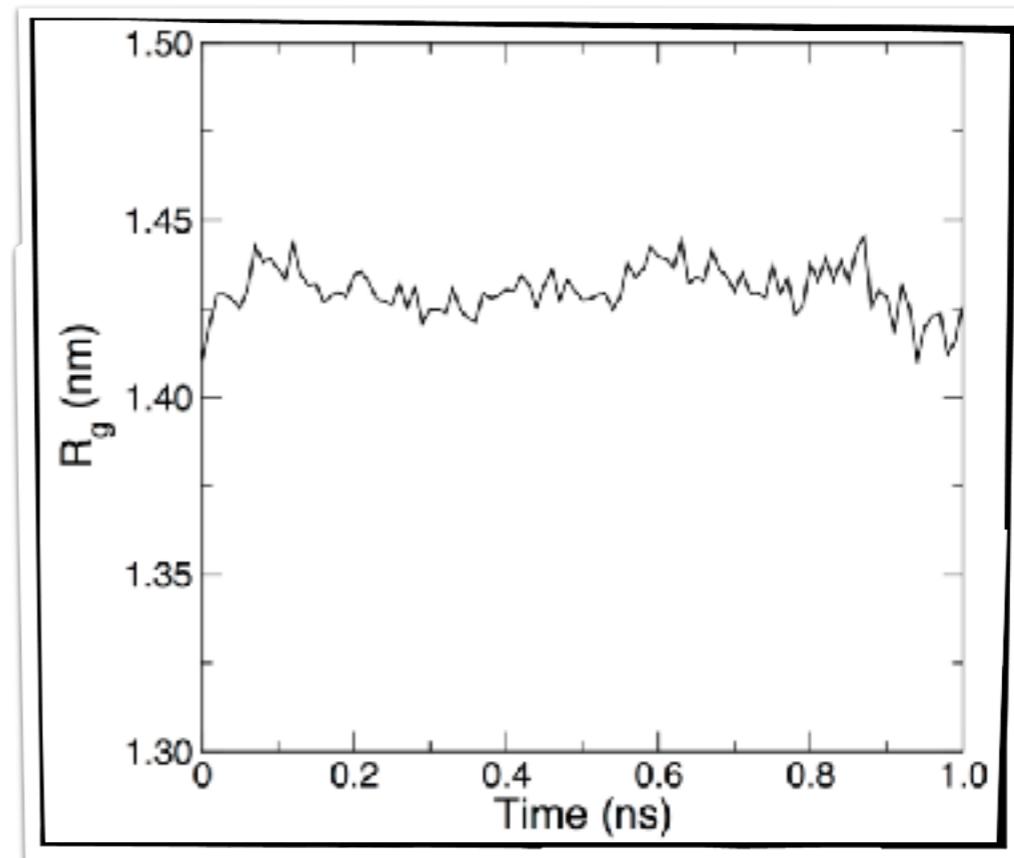
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

## Raio de giração (RG)

Descreve a compactação da estrutura proteína de estudo ao longo da simulação



```
gmx gyrate -s md.tpr -f md_noPBC.xtc -o gyrate.xvg
```

Input

Output

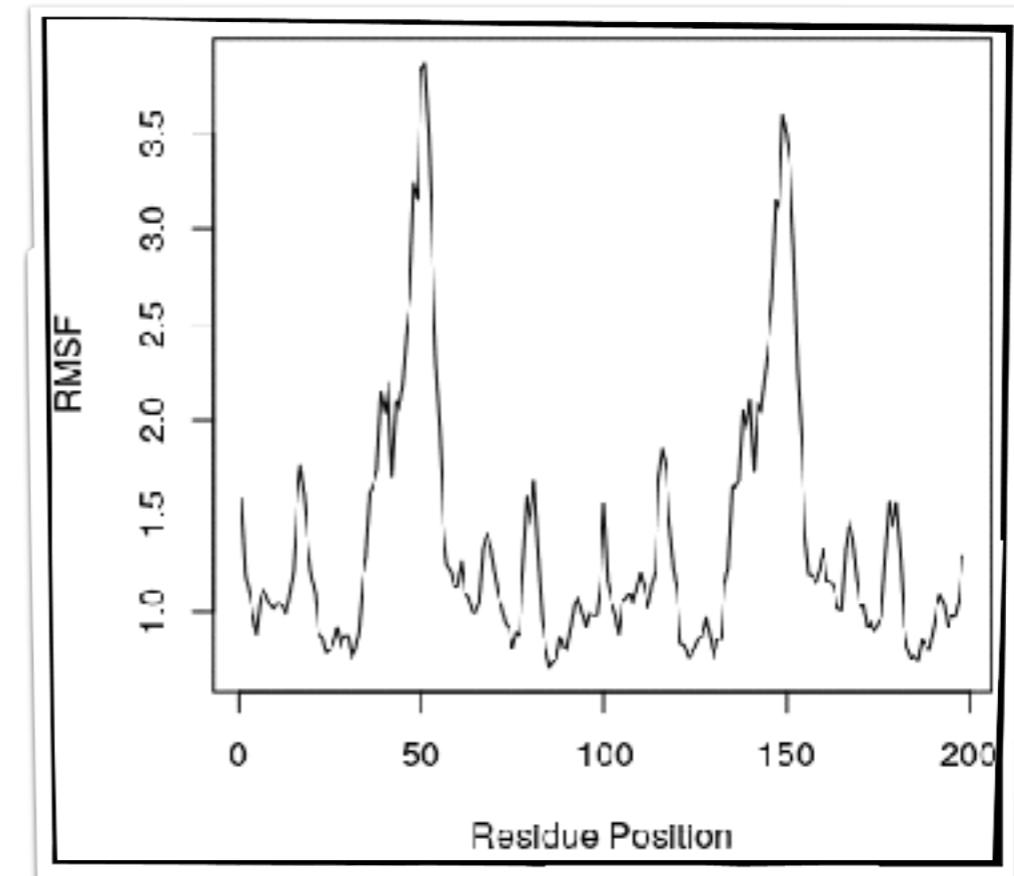
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

## Flutuação média quadrática (RMSF)

Descreve a flutuação de cada resíduo durante a simulação, fornece um indício sobre a flexibilidade de regiões na proteína



```
gmx rmsf -s md.tpr -f md_noPBC.xtc -o rmsf
```

Input

Output

# Preparação do Sistema

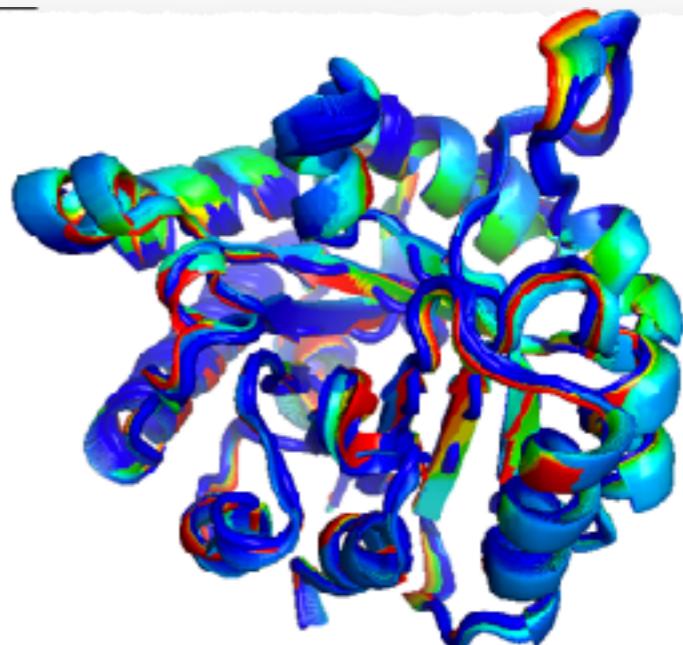
Os próximos passos na preparação de um sistema proteico para a simulação por DM

## Análise dos componentes principais (PCA)

```
gmx covar -f md_noPBC.pdb -s md.tpr
```

```
gmx ana eig -f md_noPBC.pdb -v eigenvec.trr -eig eigenval.xvg -s md.gro  
-first 1 -last 3 -filt filtered.pdb -rmsf rmsf_pca
```

## MdLovoFit



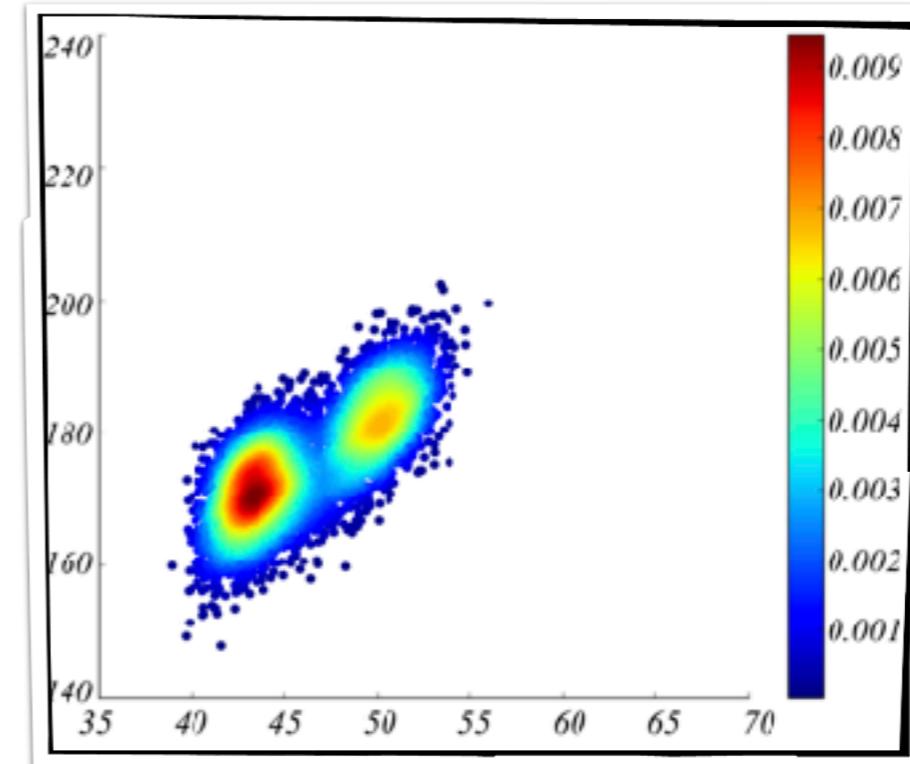
# Preparação do Sistema

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação



*Pacote de análises desenvolvido  
em python 2.7*



The "Geometric Analysis" script that was developed to carry out geometric analysis on protein structures. This script support as input MultiPDB and XTC files, and the available analysis are:

- 1 - Pincer angle.
- 2 - Dihedral angle.
- 3 - Triangle area.
- 4 - RMSD
- 5 - RG

# **Simulação por meio de dinâmica molecular clássica de um sistema *proteína-ligante***

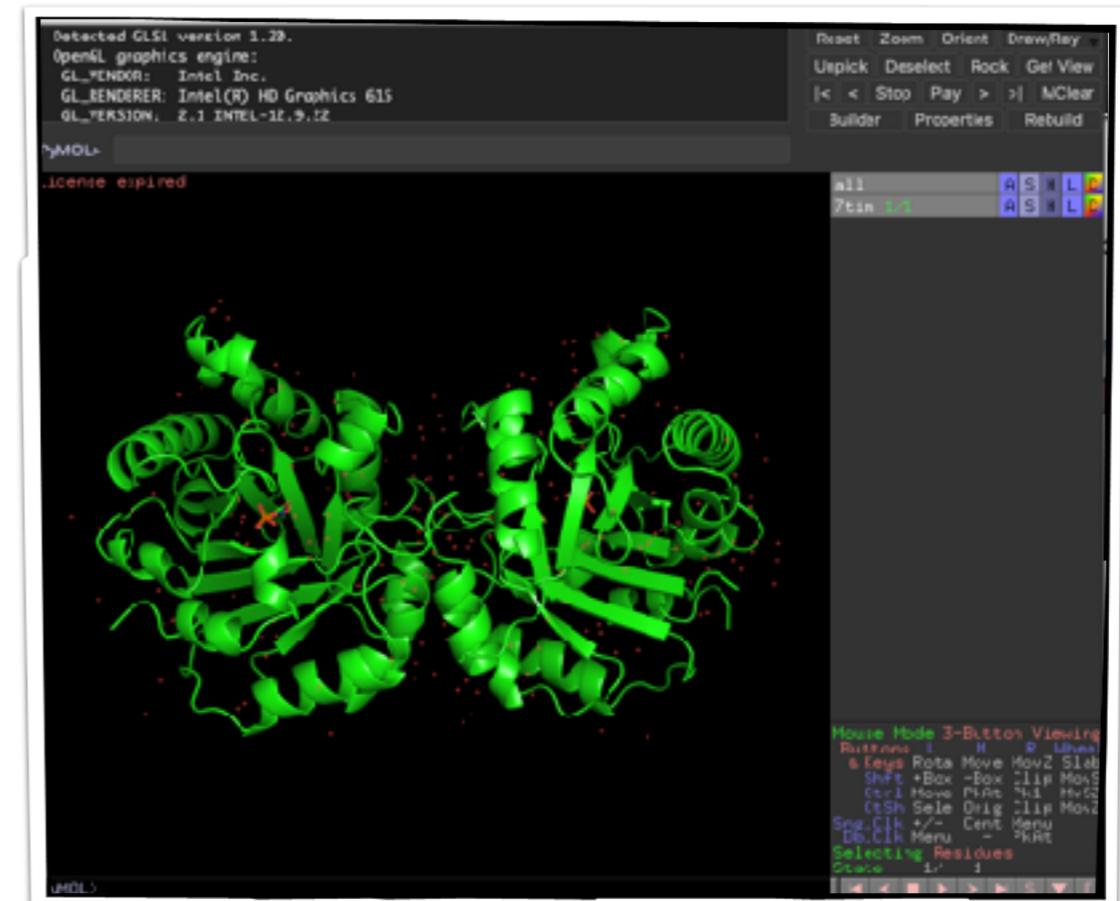
# Preparação do Sistema Proteína-Ligante

Para realizar uma simulação precisamos de um arquivo PDB da proteína de interesse

→ Neste tutorial utilizaremos a proteína [Triosefosfato Isomerase](#) (código PDB: [7TIM](#))

## *Editar o arquivo no PyMOL*

## *Acessar o site do PDB*



Remover as H<sub>2</sub>O, deletar uma cadeia

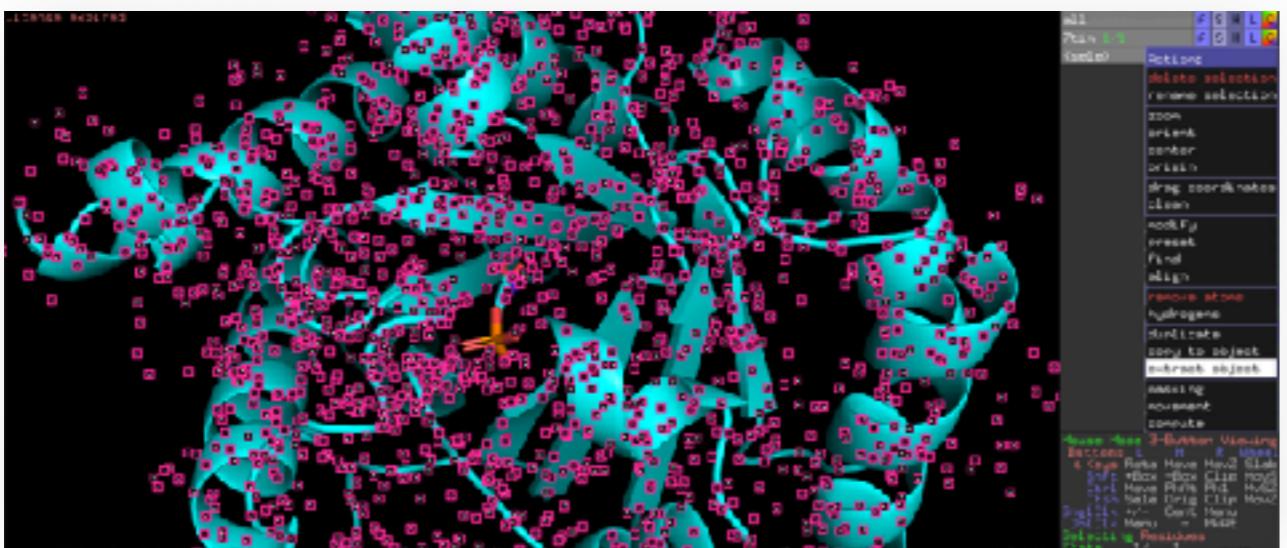
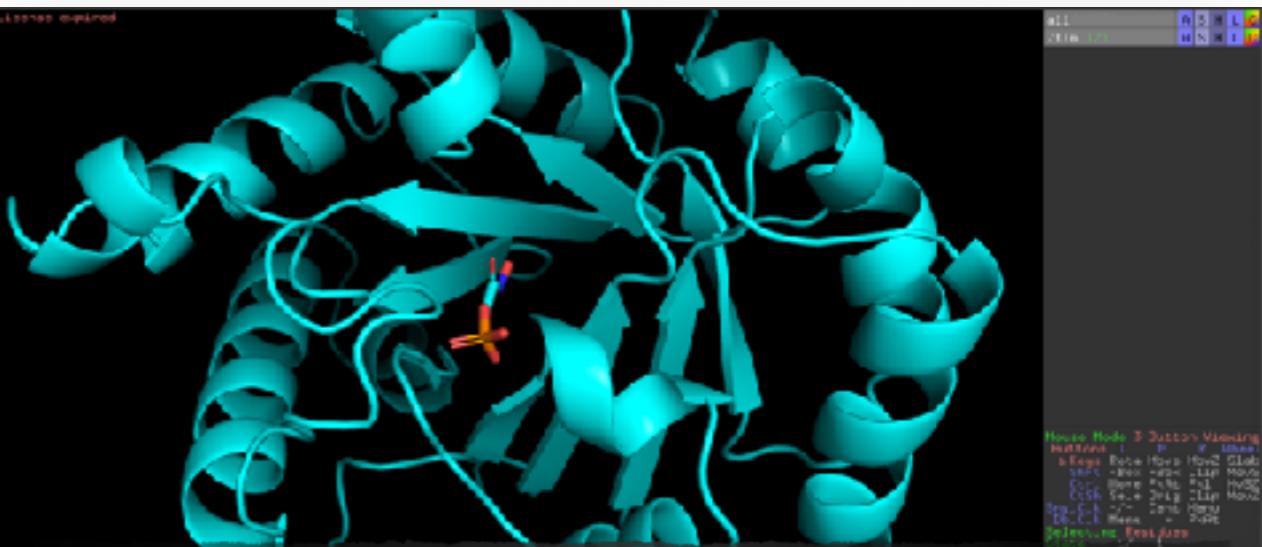
# Preparação do Sistema

Para realizar uma simulação precisamos de um arquivo PDB da proteína de interesse

→ Triosefósfato Isomerase (código PDB: **7TIM**)

Editando o arquivo PDB no PyMOL

Orientar a molécula no centro:  
A > orient



Selecionar apenas a proteína e extrair os átomos para um objeto:  
...na seleção (sele) - “A > extract object”

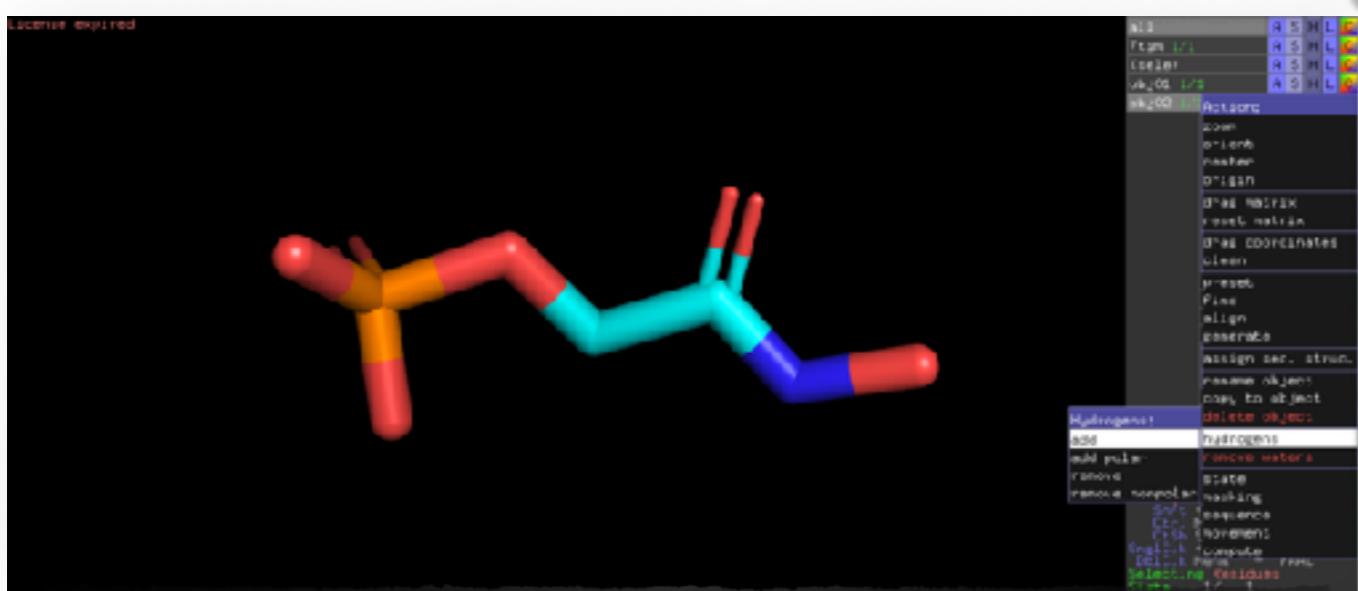
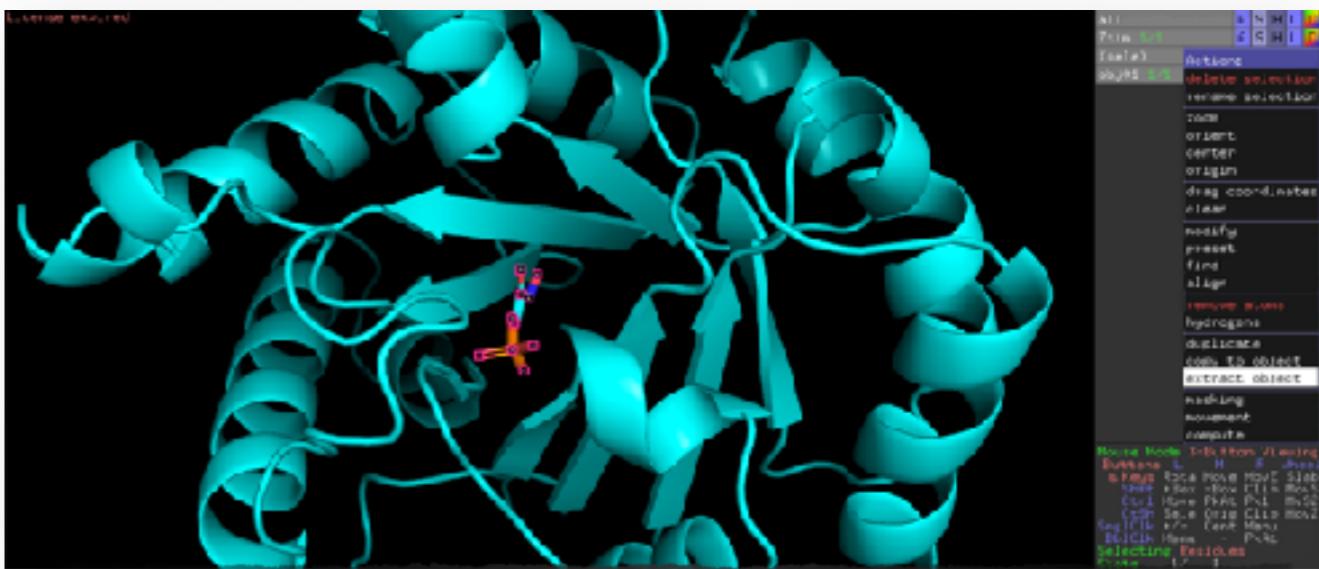
# Preparação do Sistema

Para realizar uma simulação precisamos de um arquivo PDB da proteína de interesse

→ Triosefósfato Isomerase (código PDB: **7TIM**)

*Editando o arquivo PDB no PyMOL*

Selecionar apenas o ligante e extrair os átomos para um objeto:  
...na seleção (sele) - “A > extract object”



Adicionar os hidrogênios no ligante:  
*A > hydrogens > add*

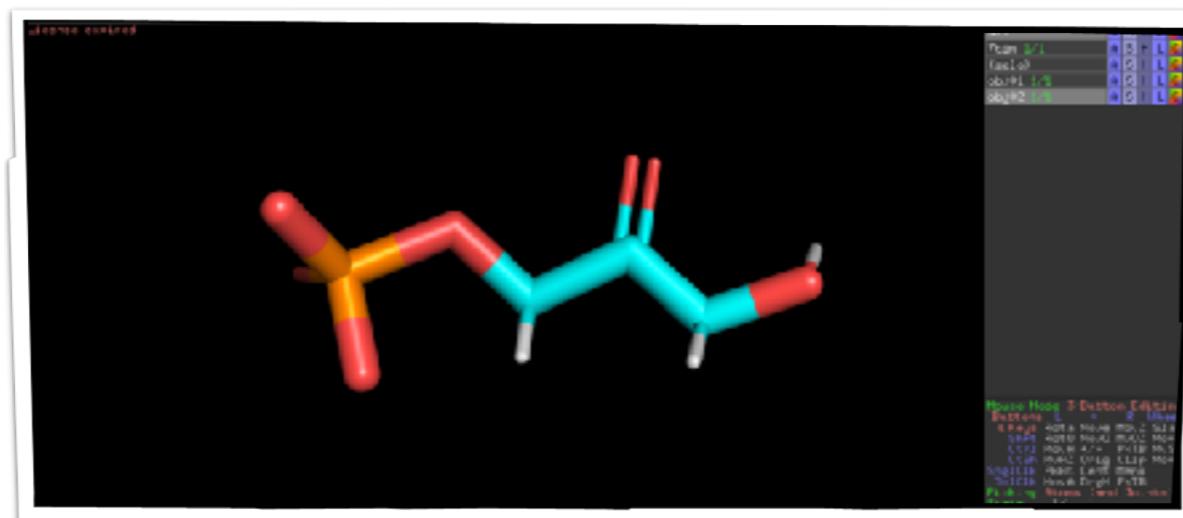
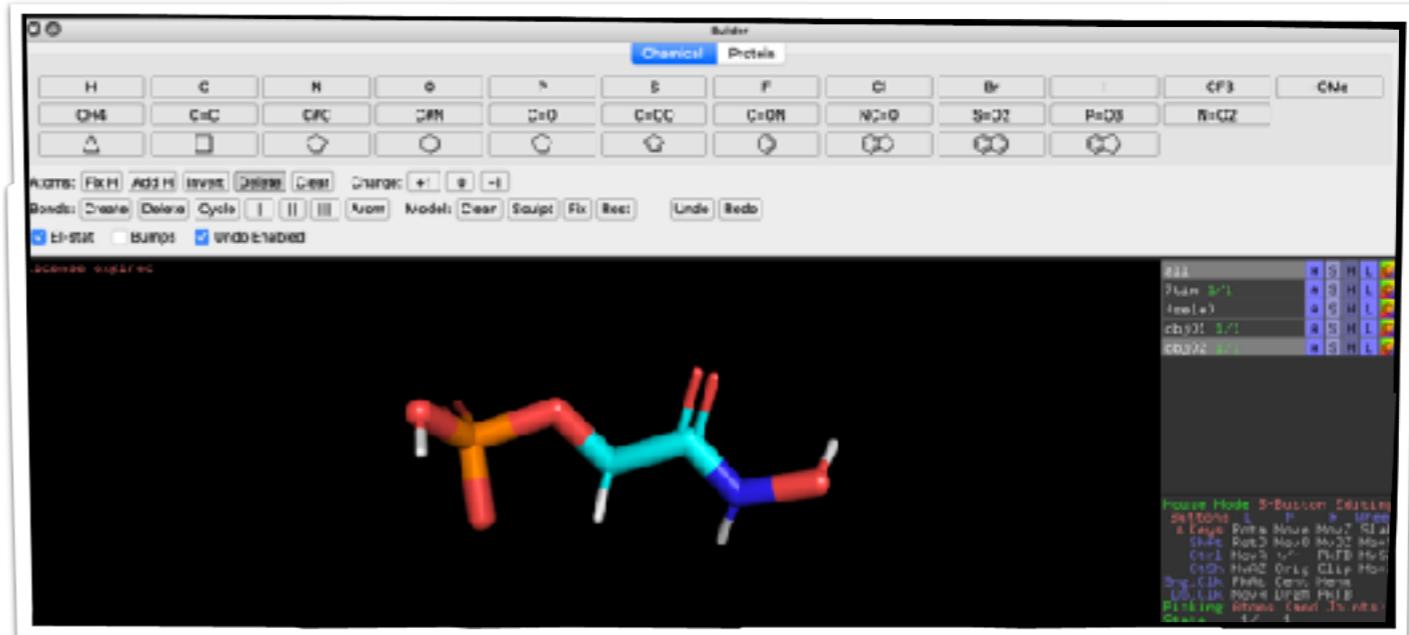
# Preparação do Sistema

Para realizar uma simulação precisamos de um arquivo PDB da proteína de interesse

→ Triosefósfato Isomerase (código PDB: **7TIM**)

## Editando o arquivo PDB no PyMOL

Acionar a função “*builder*” e deletar o “H” ligado ao oxigênio do grupo fosfato.  
Alterar o átomo de “Nitrogênio” por “Carbono”



Salvar o ligante com a extensão “.mol2”

# Preparação do Sistema

A primeira etapa para a DM é a preparação do arquivo PDB conforme os parâmetros do campo de força selecionado.



A partir do arquivo PDB serão gerados **três** arquivos:

- 1. Arquivo de topologia (.top)
- 2. Arquivo de restrição de posições (posre.itp)
- 3. Arquivo de estrutura processado (.gro)

Desta vez iremos optar por um arquivo de saída com extensão “.pdb”. Assim será mais fácil para as próximas etapas!

```
gmx pdb2gmx -f protein.pdb -o protein2.pdb -p protein.top -water tip3p
```



# Preparação do Sistema

A primeira etapa após salvar os arquivos da proteína (.pdb) e do ligante (.mol2) no PyMOL é parametrizar o ligante de acordo com o campo de força que será utilizado na simulação

Nesta etapa iremos utilizar o programa **ACPYPE**

- Caso o link para o ACPYPE não esteja pronto, digite no terminal:

ln -s \$PWD/acpype.py /usr/local/bin/acpype



- Após as etapas anteriores, seguir com a parametrização do ligante:

```
$ acpype -i ligand.mol2 -n -2
```

# Preparação do Sistema

A primeira etapa após salvar os arquivos da proteína (.pdb) e do ligante (.mol2) no PyMOL é parametrizar o ligante de acordo com o campo de força que será utilizado na simulação

*Nesta etapa iremos utilizar o programa **ACPYPE***

```
=====
| ACPYPE: AnteChamber PYthon Parser interfacE v. 2019-07-10T18:04:00UTC (c) 2019 AWSdS |
=====

==> ... charge set to -2
==> Executing Antechamber...
==> AC output file present... doing nothing
==> * Antechamber OK *
==> * Parmchk OK *
==> Topologies files already present... doing nothing
==> * Tleap OK *
==> Removing temporary files...
==> Writing NEW PDB file

==> Writing CNS/XPLOR files

==> Writing GROMACS files

==> Writing GMX dihedrals for GMX 4.5 and higher.

==> Writing CHARMM files

==> Pickle file ligand.pkl already present... doing nothing
Total time of execution: 1s
```

# Preparação do Sistema

A primeira etapa após salvar os arquivos da proteína (.pdb) e do ligante (.mol2) no PyMOL é parametrizar o ligante de acordo com o campo de força que será utilizado na simulação

Nesta etapa iremos utilizar o programa **ACPYPE**

- O ACPYPE irá criar uma pasta (ex.: “ligand.acpype”) com diversos arquivos de parâmetros para os campos de força AMBER, CNS e OPLS-AA

```
# Adicionar o ligante gerado com o ACPYPE no arquivo ".pdb" da proteína:
```

```
grep -h ATOM Protein2.pdb Ligand.acpype/Ligand_NEW.pdb >| Complex.pdb
```

Este comando irá colocar o arquivo novo do ligante junto com o arquivo da proteína gerado com o módulo “pdb2gmx”

- Parte final do arquivo “complex.pdb”

ATOM	3775	ZHD2	ASN	B	248	38.450	31.879	49.565	1.00	0.00	C
ATOM	3776	C	ASN	B	248	38.307	28.868	45.077	1.00	0.00	O
ATOM	3777	OC1	ASN	B	248	38.660	28.884	45.045	1.00	0.00	O
ATOM	3778	OC2	ASN	B	248	37.718	28.727	44.160	1.00	0.00	O
ATOM	1	P	PGH	Z	1	54.043	41.877	55.729	1.00	0.00	P
ATOM	2	O1P	PGH	Z	1	55.348	42.237	54.795	1.00	0.00	O
ATOM	3	O2P	PGH	Z	1	54.733	41.344	56.995	1.00	0.00	O
ATOM	4	O3P	PGH	Z	1	53.348	40.921	54.953	1.00	0.00	O
ATOM	5	O4P	PGH	Z	1	53.352	43.194	56.005	1.00	0.00	O
ATOM	6	C01	PGH	Z	1	56.350	42.824	51.381	1.00	0.00	C
ATOM	7	C1	PGH	Z	1	56.324	42.574	52.595	1.00	0.00	C
ATOM	8	O1	PGH	Z	1	57.161	43.351	53.110	1.00	0.00	O
ATOM	9	C2	PGH	Z	1	55.103	42.885	53.385	1.00	0.00	C
ATOM	10	O2	PGH	Z	1	57.403	42.259	50.452	1.00	0.00	O
ATOM	11	H01	PGH	Z	1	57.576	43.200	50.375	1.00	0.00	H
ATOM	12	H03	PGH	Z	1	54.921	41.834	53.159	1.00	0.00	H
ATOM	13	H02	PGH	Z	1	55.433	42.342	50.685	1.00	0.00	H
ATOM	14	H04	PGH	Z	1	56.498	40.970	51.619	1.00	0.00	H
ATOM	15	H05	PGH	Z	1	54.230	42.674	53.104	1.00	0.00	H

# Preparação do Sistema

A primeira etapa após salvar os arquivos da proteína (.pdb) e do ligante (.mol2) no PyMOL é parametrizar o ligante de acordo com o campo de força que será utilizado na simulação

Nesta etapa iremos utilizar o programa **ACPYPE**

- Dentro da pasta que o ACPYPE criou precisamos mover o arquivo “ligand\_GMX.itp” para a pasta que estamos montando o sistema

No arquivo “protein.top”, realizar duas alterações

- No final do arquivo:

```
[ molecules ]
; Compound          #mols
Protein_chain_B    1
PGH                1
```

- No início do arquivo:

```
; Include forcefield parameters
#include "amber99sb.ff/forcefield.itp"
#include "ligand_GMX.itp"
```



- Alterar:

```
[ moleculetype ]
;name          nrexcl
ligand         3
```

- Para:

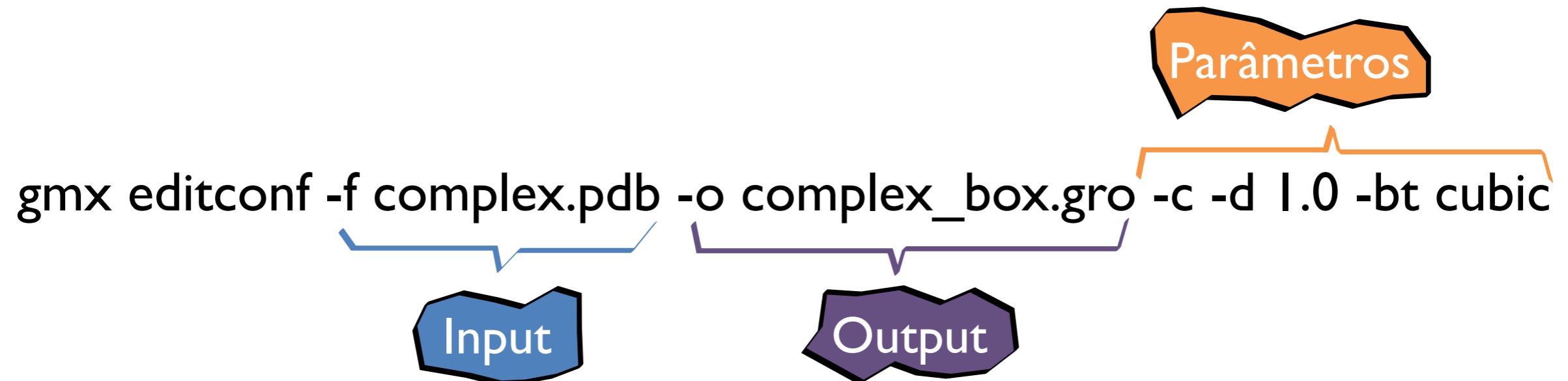
```
[ moleculetype ]
;name          nrexcl
PGH           3
```

# Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

→ (i) Definir as dimensões da caixa por meio do módulo “**editconf**”

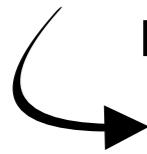
- **c**: Centralizar a proteína no centro da caixa
- **d**: Distância da proteína até as arestas da caixa
- **bt**: Tipo de caixa que será utilizada



# Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

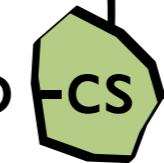
(ii) Preencher a caixa gerada com as moléculas por meio do módulo “**solvate**”



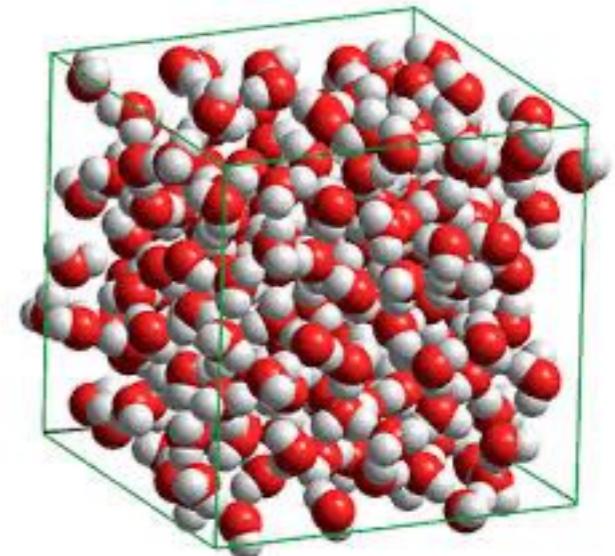
- *Configuração dos parâmetros de solvente*

```
gmx solvate -cp complex_box.gro -p protein.top -cs -o complex_solv.gro
```

Input



Output



# Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

- Adicionar os íons com o intuito de neutralizar as cargas do sistema de estudo

```
gmx grompp -f ionsmdp -c complex_solv.gro -p protein.top -o ions.tpr
```

Input

Output

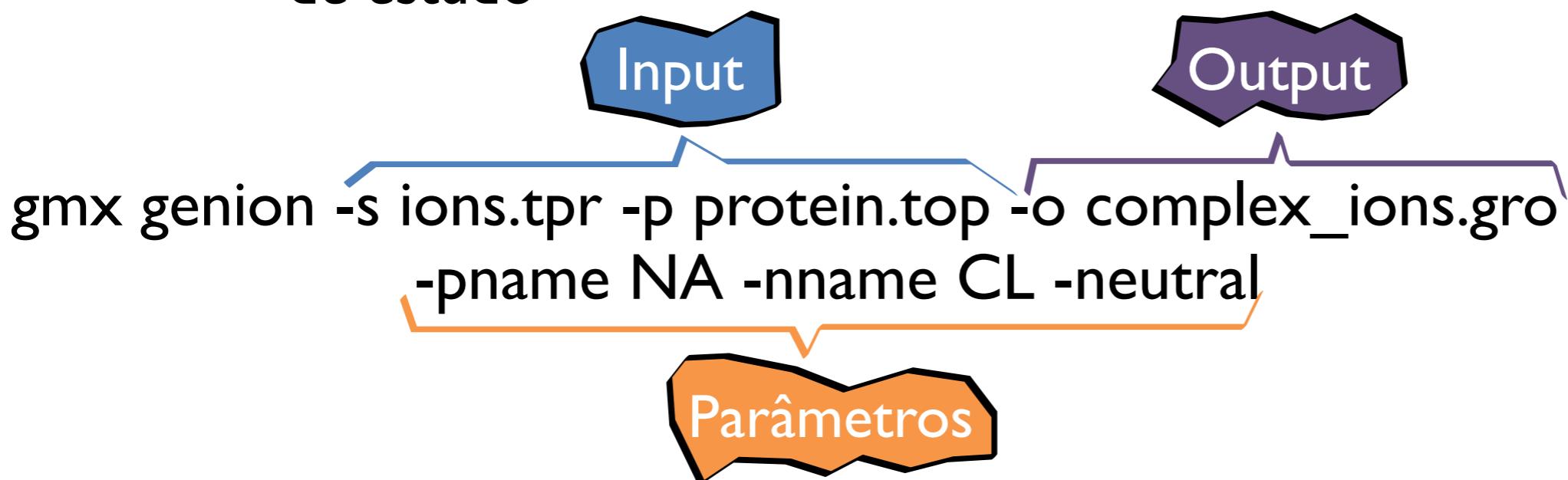
```
NOTE 2 [file protein_2.top, line 35614]:  
System has non-zero total charge: -5.000002  
Total charge should normally be an integer. See  
http://www.gromacs.org/Documentation/Floating\_Point\_Arithmetic  
for discussion on how close it should be to an integer.
```

Descreve a carga do sistema, informando quantos íons deverão ser adicionados

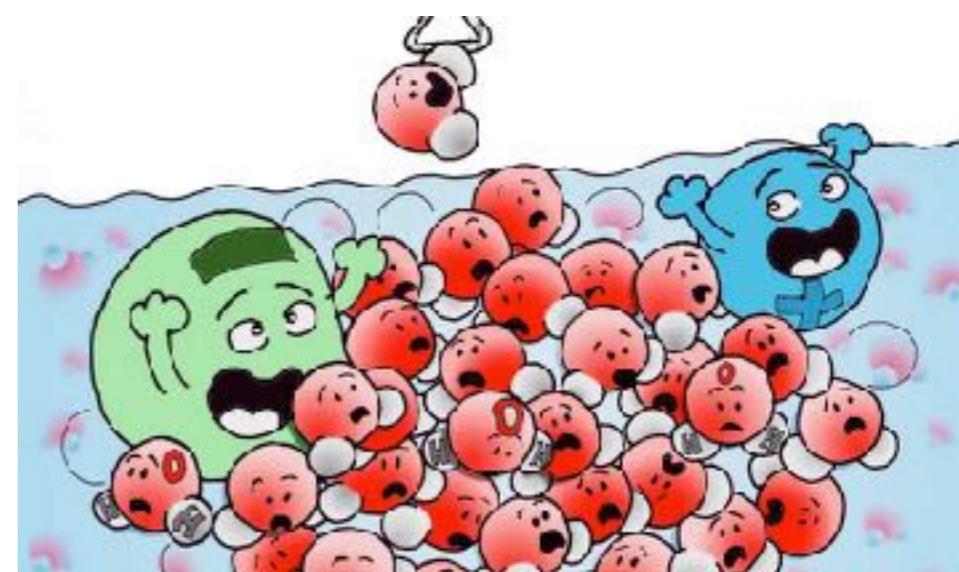
# Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

→ Adicionar os íons com o intuito de neutralizar as cargas do sistema de estudo



Determina que serão adicionados o número exato de íons para neutralizar o sistema. Existe a possibilidade de adicionar os íons de acordo com uma concentração específica, i.e. “**-conc 0.15 -neutral**”



# Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

→ Adicionar os íons com o intuito de neutralizar as cargas do sistema de estudo

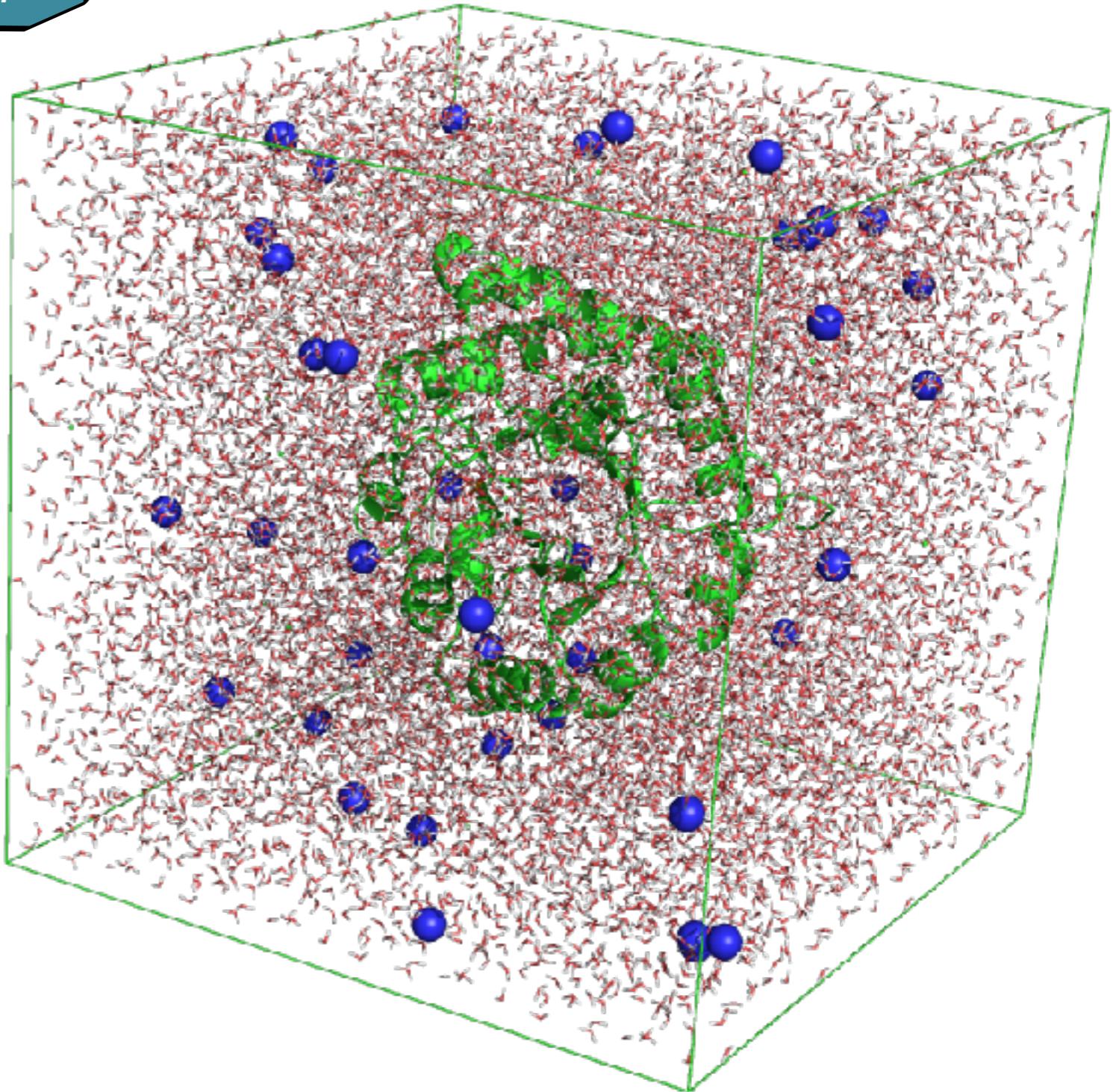
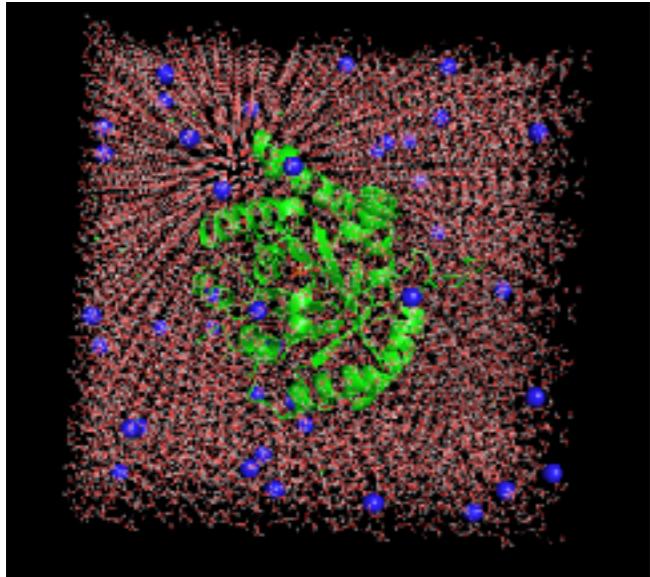
```
Reading file ions.tpr, VERSION 2018.1 (single precision)
Reading file ions.tpr, VERSION 2018.1 (single precision)
Will try to add 5 NA ions and 0 CL ions.
Select a continuous group of solvent molecules
Group      0 (      System) has  38578 elements
Group      1 (      Protein) has   3778 elements
Group      2 (  Protein-H) has   1883 elements
Group      3 (    C-alpha) has    247 elements
Group      4 (    Backbone) has   741 elements
Group      5 (   MainChain) has   989 elements
Group      6 ( MainChain+Cb) has  1214 elements
Group      7 ( MainChain+H) has  1231 elements
Group      8 (   SideChain) has  2547 elements
Group      9 ( SideChain-H) has   894 elements
Group     10 (  Prot-Masses) has  3778 elements
Group     11 ( non-Protein) has 34800 elements
Group     12 (      Other) has    15 elements
Group     13 (       PGH) has    15 elements
Group     14 (      Water) has 34785 elements
Group     15 (        SOL) has 34785 elements
Group     16 ( non-Water) has  3793 elements
Select a group: 15
```

Selecionar o grupo que esteja o solvete! Serão trocadas moléculas de água por íons!!!

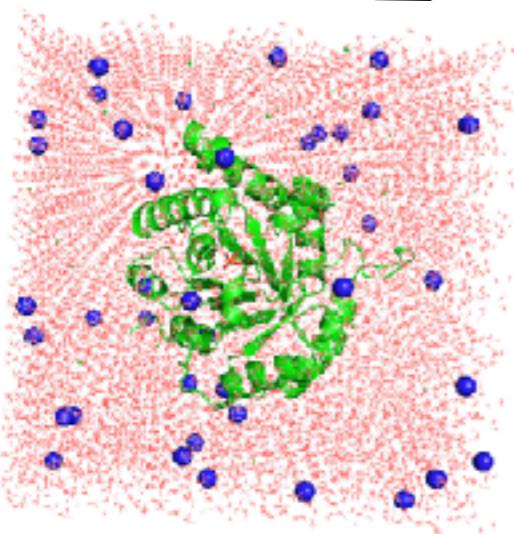
# Preparação do Sistema

Visualização do sistema no **PyMOL**

...digitar no terminal: **pymol protein\_ions.pdb**



**>bg\_color white**



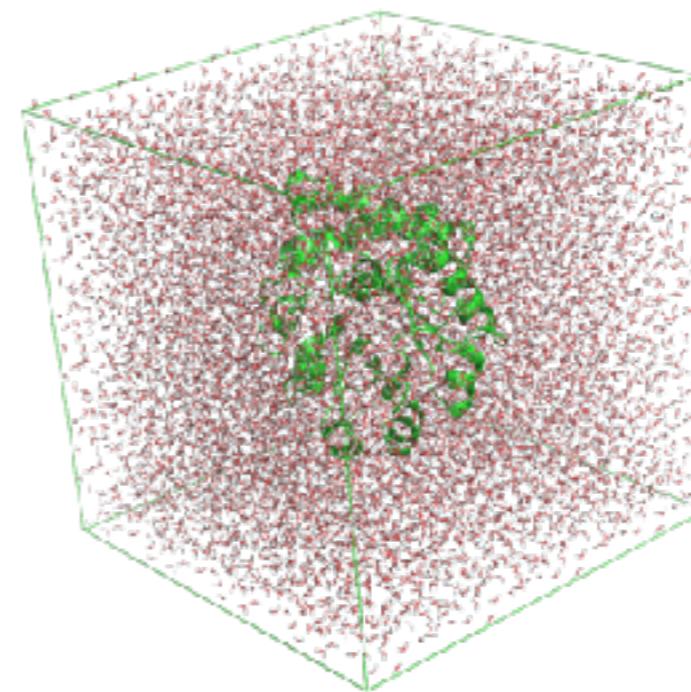
## Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

→ Adicionar os ions com o intuito de neutralizar as cargas do sistema de estudo

...conferindo o arquivo de simulação...

Converte o arquivo “.gro” em  
“.pdb” para a visualização  
no PyMOL



`gmx editconf -f complex_ions.gro -o complex_ions.pdb`

Input

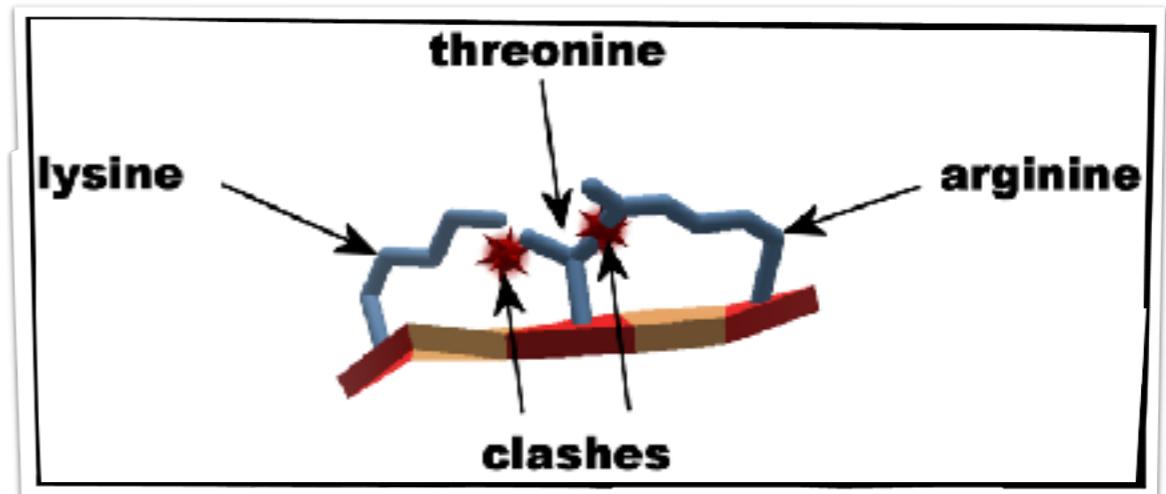
Output

# Preparação do Sistema

Após a parametrização do ligante, a preparação do sistema segue da mesma forma quando comparado com a simulação livre

→ Minimização do sistema para evitar geometrias incorretas, assim como, choques entre cadeias laterais do resíduos

Etapa de pré-  
processamento



gmx grompp -f minim.mdp -c complex\_ions.gro -p protein.top -o em.tpr

Input

Output

This run will generate roughly 3 Mb of data

→ Fornece uma estimativa de espaço!

# Preparação do Sistema

Os passos seguintes são...

...opção para impacientes!

```
gmx mdrun -deffnm em -v
```

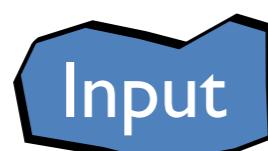


```
gmx grompp -f nvtmdp -c em.gro -r em.gro -p protein.top -o nvt.tpr
```



...opção para impacientes!

```
gmx mdrun -deffnm nvt -v
```



```
gmx grompp -f nptmdp -c nvt.gro -r nvt.gro -t nvt.cpt -p protein.top -o npt.tpr
```



# Preparação do Sistema

**Os passos seguintes são...**

*...opção para impacientes!*

```
gmx mdrun -deffnm npt -v
```



```
gmx grompp -f md.mdp -c npt.gro -r npt.gro -t npt cpt -p protein.top -o md.tpr
```



*...opção para impacientes!*

```
gmx mdrun -deffnm md -v
```



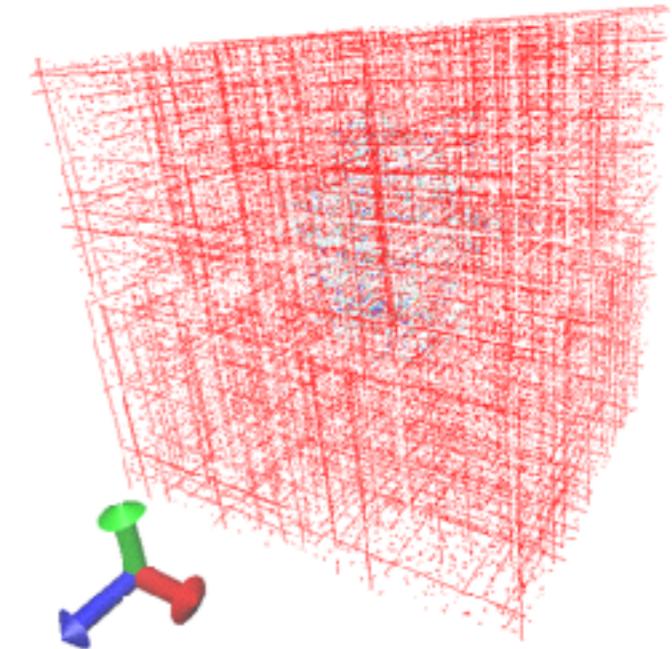
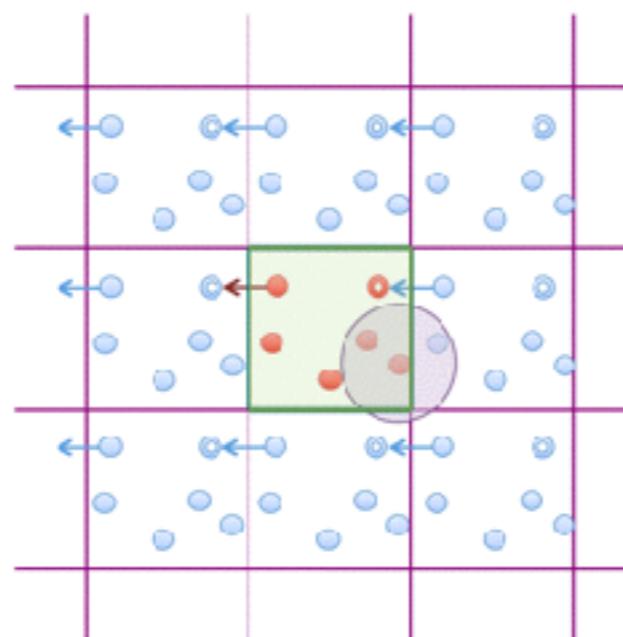
# Preparação do Sistema - Análises

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

Condições periódicas de contorno (PBC)

A partir deste ponto iremos centralizar todos os frames da proteína no centro da caixa do sistema



```
gmx trjconv -s md.tpr -f md.xtc -pbc mol -center -o md_noPBC.xtc
```

Input

Parâmetros

Output

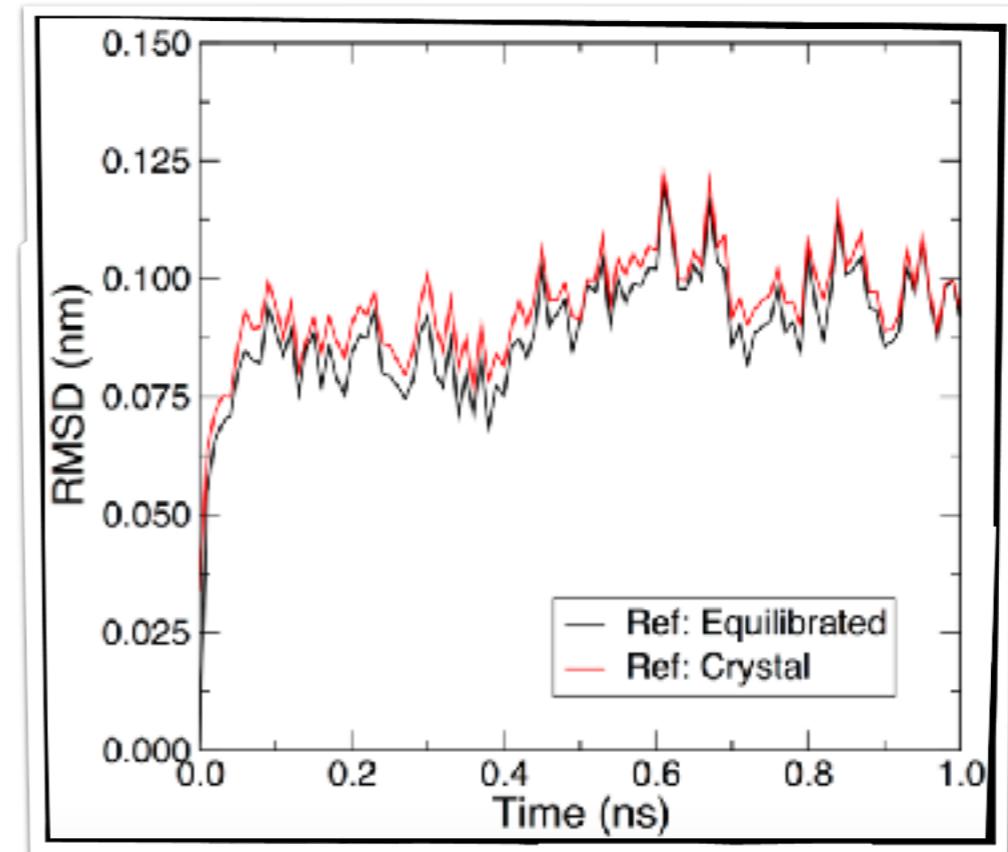
# Preparação do Sistema - Análises

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

## Desvio médio quadrático (RMSD)

Descreve a variação da estrutura da proteína ao longo da simulação em relação a estrutura inicial



```
gmx rms -s md.tpr -f md_noPBC.xtc -o rmsd.xvg -tu ns
```



Input

Output

Parâmetros

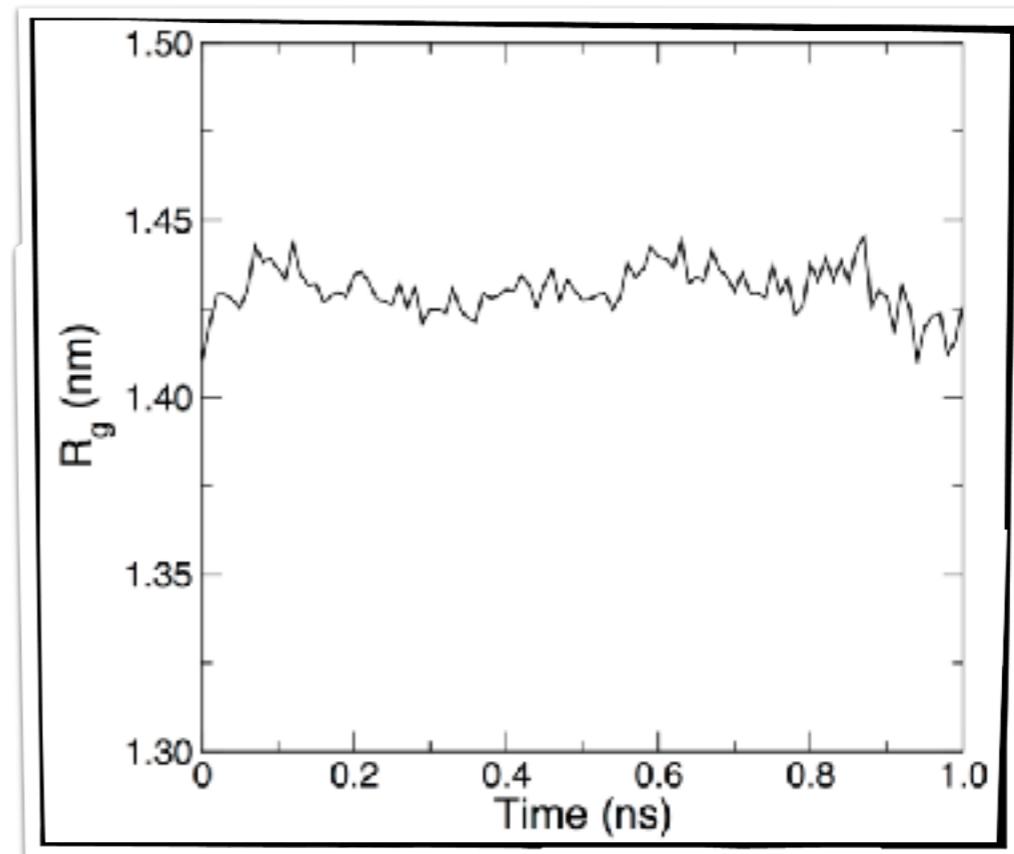
## Preparação do Sistema - Análises

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação

### Raio de giração (RG)

Descreve a compactação da estrutura proteína de estudo ao longo da simulação



```
gmx gyrate -s md.tpr -f md_noPBC.xtc -o gyrate.xvg
```

Input

Output

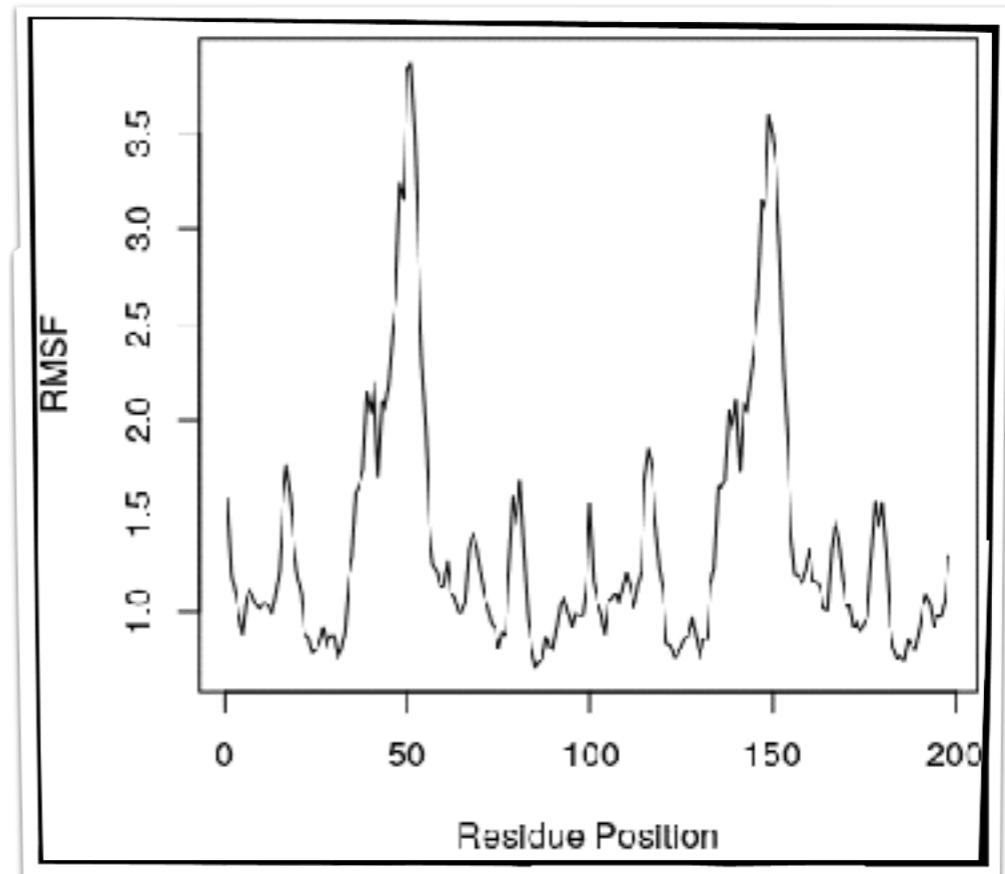
# Preparação do Sistema - Análises

Os próximos passos na preparação de um sistema proteico para a simulação por DM

Após a simulação, podemos iniciar as etapas de análises da simulação

## Flutuação média quadrática (RMSF)

Descreve a flutuação de cada resíduo durante a simulação, fornece um indício sobre a flexibilidade de regiões na proteína



gmx rmsf -s md.tpr -f md\_noPBC.xtc -o rmsf

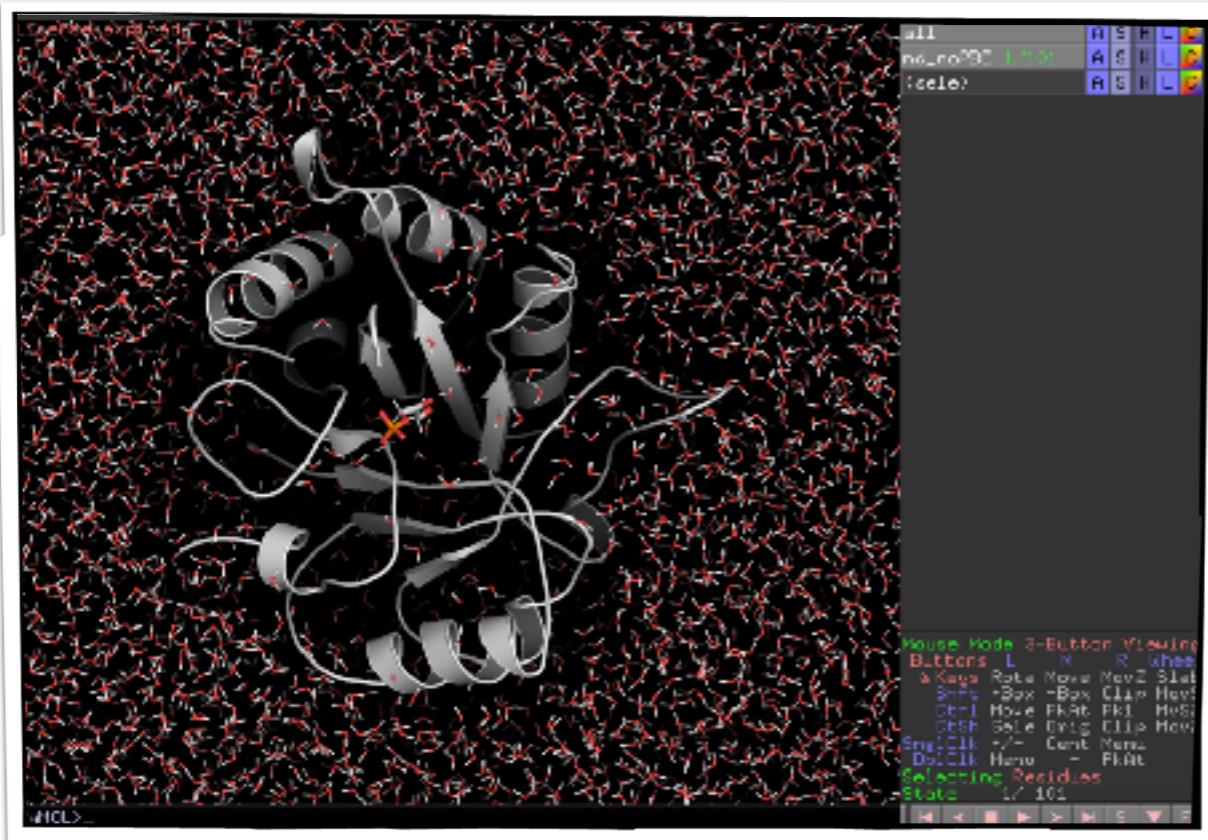
Input

Output

## Preparação do Sistema - Análises

Os próximos passos na preparação de um sistema proteico para a simulação por DM

→ Após a simulação, podemos iniciar as etapas de análises da simulação



Abrir o arquivo “md\_noPBC.pdb”  
no PyMOL para visualizar a  
trajetória

```
gmx trjconv -s md.tpr -f md.xtc -pbc mol -center -o md_noPBC.pdb
```

Input

Parâmetros

Output

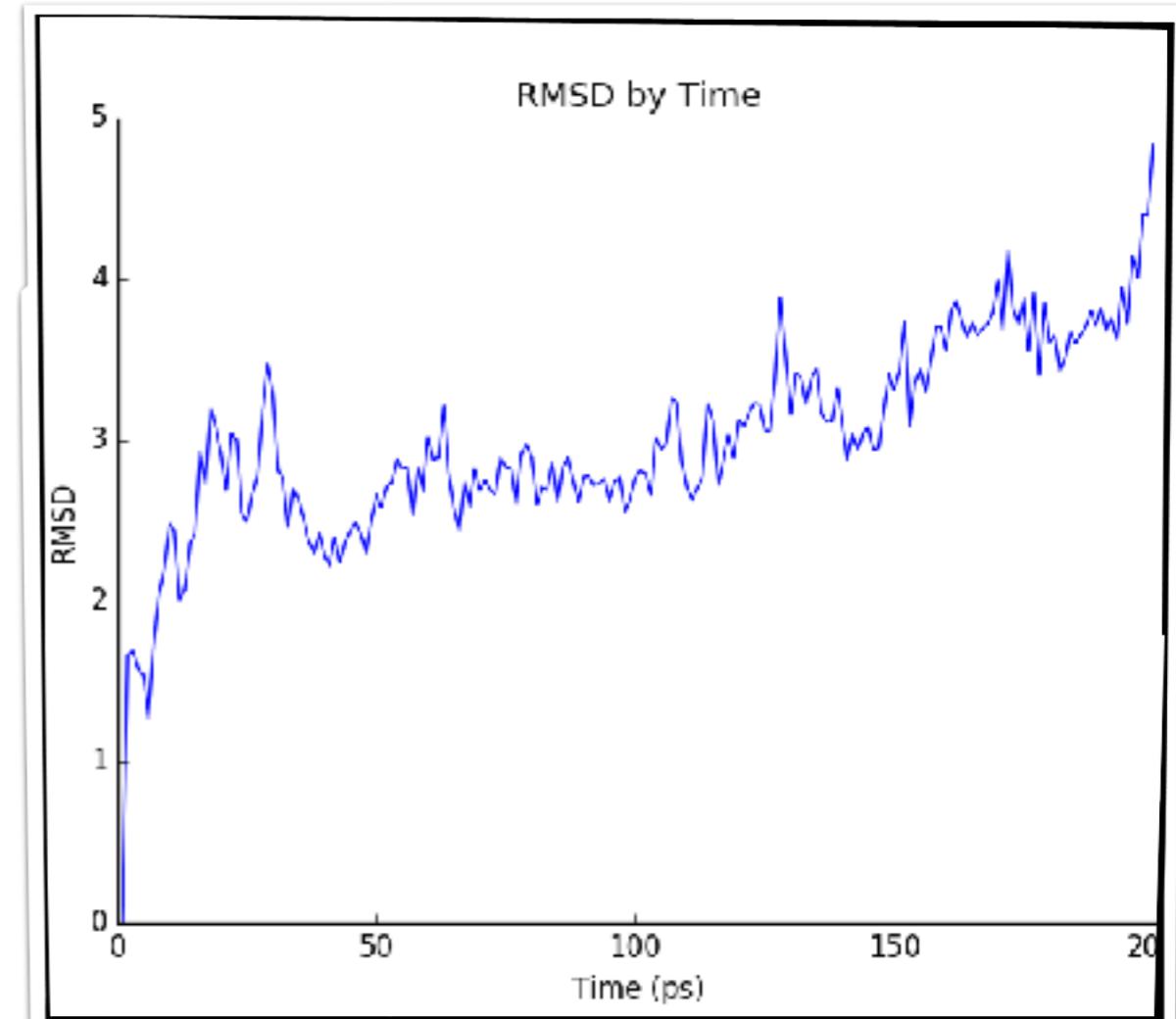
# Preparação do Sistema - Análises Gambiarra Package

Fazer o download do arquivo “teste2.tar.gz” pelo github:

[https://github.com/luisfernandosaraiva/BBMC\\_2022](https://github.com/luisfernandosaraiva/BBMC_2022)

```
python geometric_measures.py -m teste2.pdb -t all -c A -rA 1,119,226 -rD 1,119,226,34 -tA 1,119,226 -o all
```

```
python plot_geometric_measures.py -p all.tbl -t RMSD
```



# Preparação do Sistema - Análises

`python plot_geometric_measures.py -p all.tbl -t RMSD,Dist_AC`

